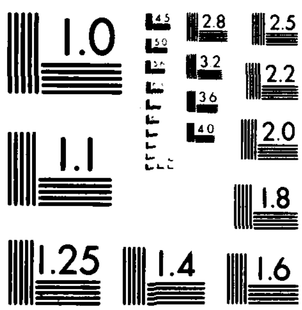
AD-A107 384 AUBURN UNIV AL DEPT OF ELECTRICAL ENGINEERING F/G 17/7
MARINE AIR TRAFFIC CONTROL AND LANDING SYSTEM (MATCALS) INVESTI--ETC(U)
APR 81 E R GRAF, C L PHILLIPS, S A STARKS N00039-80-C-0032

UNCLASSIFIED NL

1 of 2
AD
A107384



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

LEVEL II

ELECTRICAL

ENGINEERING EXPERIMENT STATION

AUBURN UNIVERSITY

AUBURN, ALABAMA

DISTRIBUTION STATEMENT A

Approved for public release; Distribution Unlimited

TECHNICAL REPORT
~~CONTRACT T-A-2550~~
(SUBCONTRACTED FROM N-00039-80-C-0032)
MARINE AIR TRAFFIC CONTROL
AND LANDING SYSTEM
(MATCALS) INVESTIGATION.

E.R. Graf, C.L. Phillips, ~~and~~ S.A. Starks
CO- PROJECT LEADERS

April, 1981

Prepared for

The Engineering Experiment Station
Georgia Institute of Technology
Atlanta, Georgia

Prepared by

The Electrical Engineering Department
Auburn University
Auburn University, Alabama

FOREWORD

This technical report is submitted to the Georgia Institute of Technology to comply with the report requirements of contract 1-A-2550, which is a subcontract under United States Navy contract N-00039-80-C-0032. This report is published in four parts, each separate and independent of the others. The final technical report of this contract is due to be submitted in September, 1981.

REPORT CONTENTS

PART ONE

REPORT SUMMARY

Prepared for

Georgia Institute of Technology
ATLANTA, GEORGIA

Under

Contract 1-A-2550

by

Electrical Engineering Department
Auburn University
Auburn, Alabama

Prepared by: Charles L. Phillips

REPORT SUMMARY

Auburn University, under contracts N66314-73-C-1565, N66314-74-C-1352, N66314-74-C-1634, N00228-75-C-2080, N00228-76-C-2069, and N00228-78-C-2233 with the United States Navy, and has investigated various aspects of the Marine Air Traffic Control and Landing System (MATCALS). This report contains the results of the continuation of these investigations under contract 1-A-2550 with the Georgia Institute of Technology. The report is organized into three main sections, namely Part Two, Part Three, and Part Four. Part Two contains the results of an investigation into replacing the α-β filter in the MATCALS digital controller with an observer, in order to reduce the effects of radar noise. Part Three presents a centroid algorithm based up return amplitude-versus-angle signature. Part Four presents an investigation of adaptive filtering algorithms for the MATCALS system.

Observer Design

Presently a problem exists in the closed-loop control of the MATCALS system due to the noise generated in AN/TPN 22 radar. An α-β filter in the flight dynamic and control module is employed to reduce the noise effects while estimating the position and the velocity of the aircraft. An observer may also be used to estimate the status of the aircraft. Part Two of this report presents the results of an investigation of the replacement of the α-β filter with an observer.

Figure 1-1 shows the F4J aircraft lateral control system containing the α-β filter. Figure 1-2 shows the same system with the α-β filter replaced by an observer. The techniques for designing an observer are simple; however, these techniques do not completely specify the observer. Certain parameters in the observer must be obtained by trial and error. The criteria used to determine these parameters are explained in Chapter 3 of Part Two.

Figure 1-3 gives a typical reponse of the lateral control system of the F4J aircraft for the final sixty seconds of flight before touchdown. The inputs to this simulation were radar noise and wind turbulence, both of which are disturbances. It is seen that the observer control system reponds less to the disturbances than does the α-β filter control system.

Table 1-1 presents the results of a Monte Carlo simulation based on twenty simulations. The column labeled r.m.s. is the root-mean-square value of the lateral displacement of the aircraft from the extended centerline of the runway for the final sixty seconds of flight. It is seen that the observer improves the system response to the radar noise, but degrades the system response to the wind turbulence.

These studies will continue. The parameters of the observer which were obtained by trial and error are probably not optimal. Hence future investigations will be directed toward a better choice of these parameters.

Radar Centroid Investigation

A method of estimating the centroid location of a target utilizing scan return amplitude-versus-angle information is introduced in Part Three. The method is compared to three thresholding estimators and a first moment estimator in a computer-simulated automatic landing system.

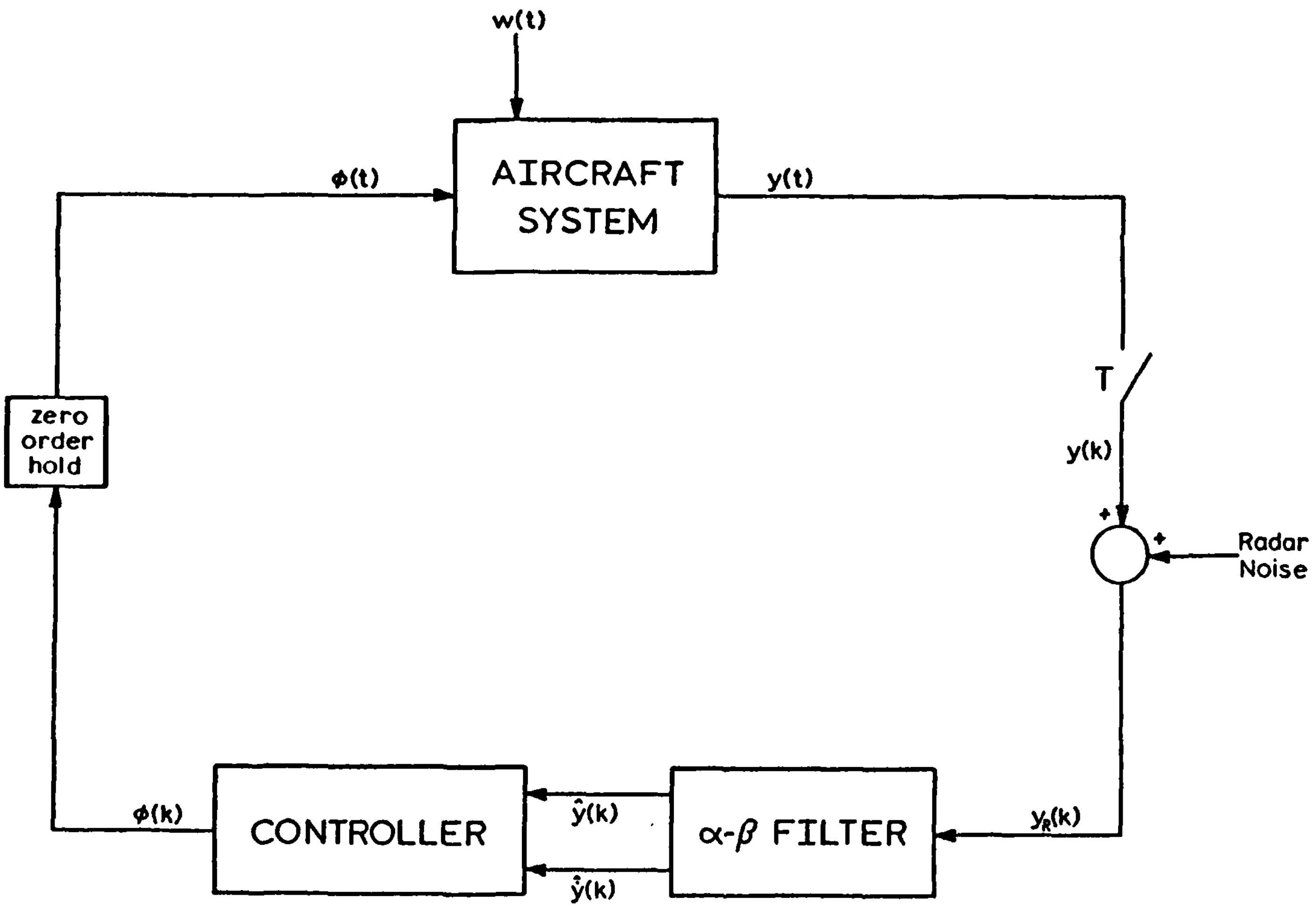


Figure 1-1. Block Diagram of the F4J Aircraft Lateral Control System with the α-β Filter.

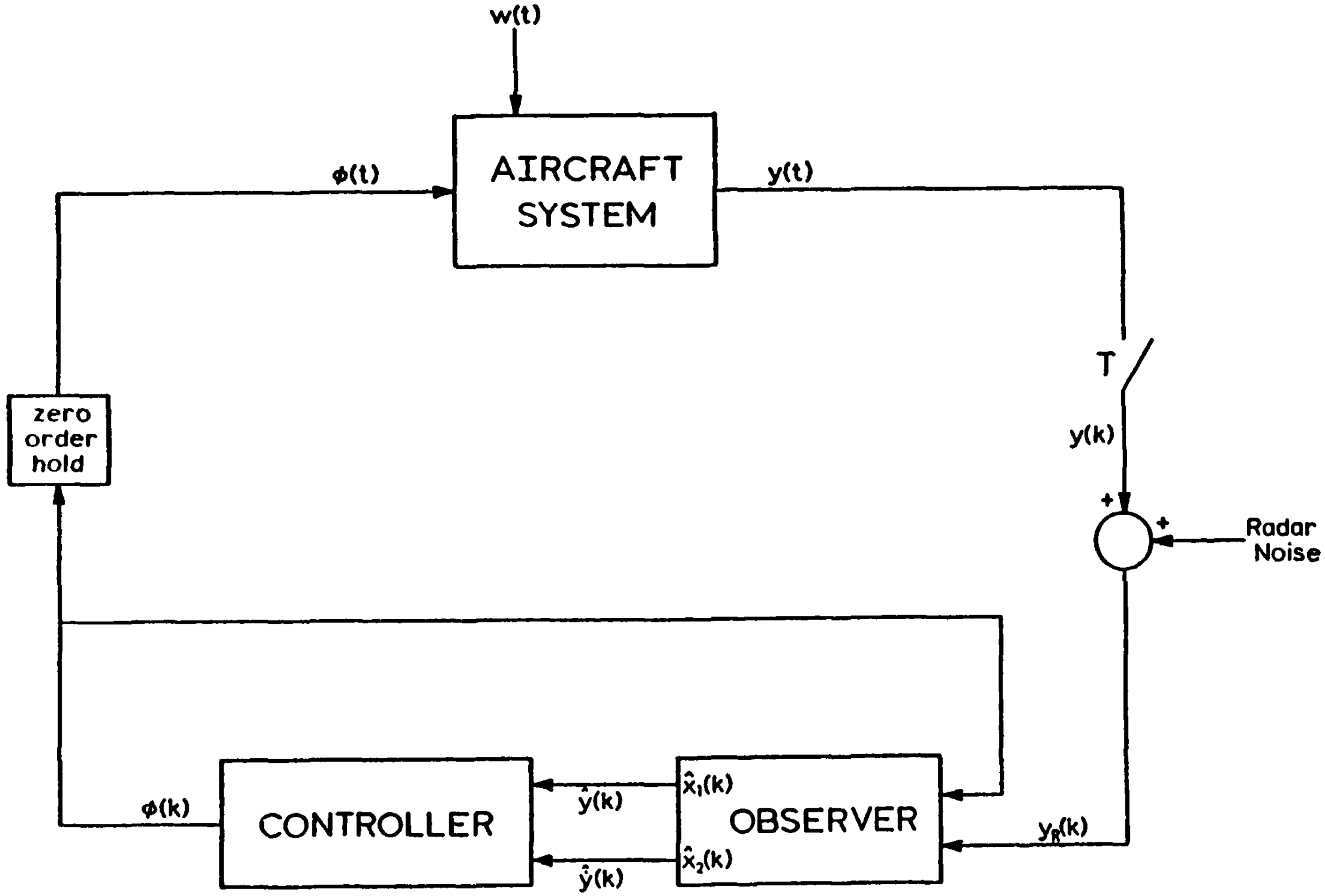


Figure 1-2. Block Diagram of the F4J Aircraft Lateral Control System with the Observer.

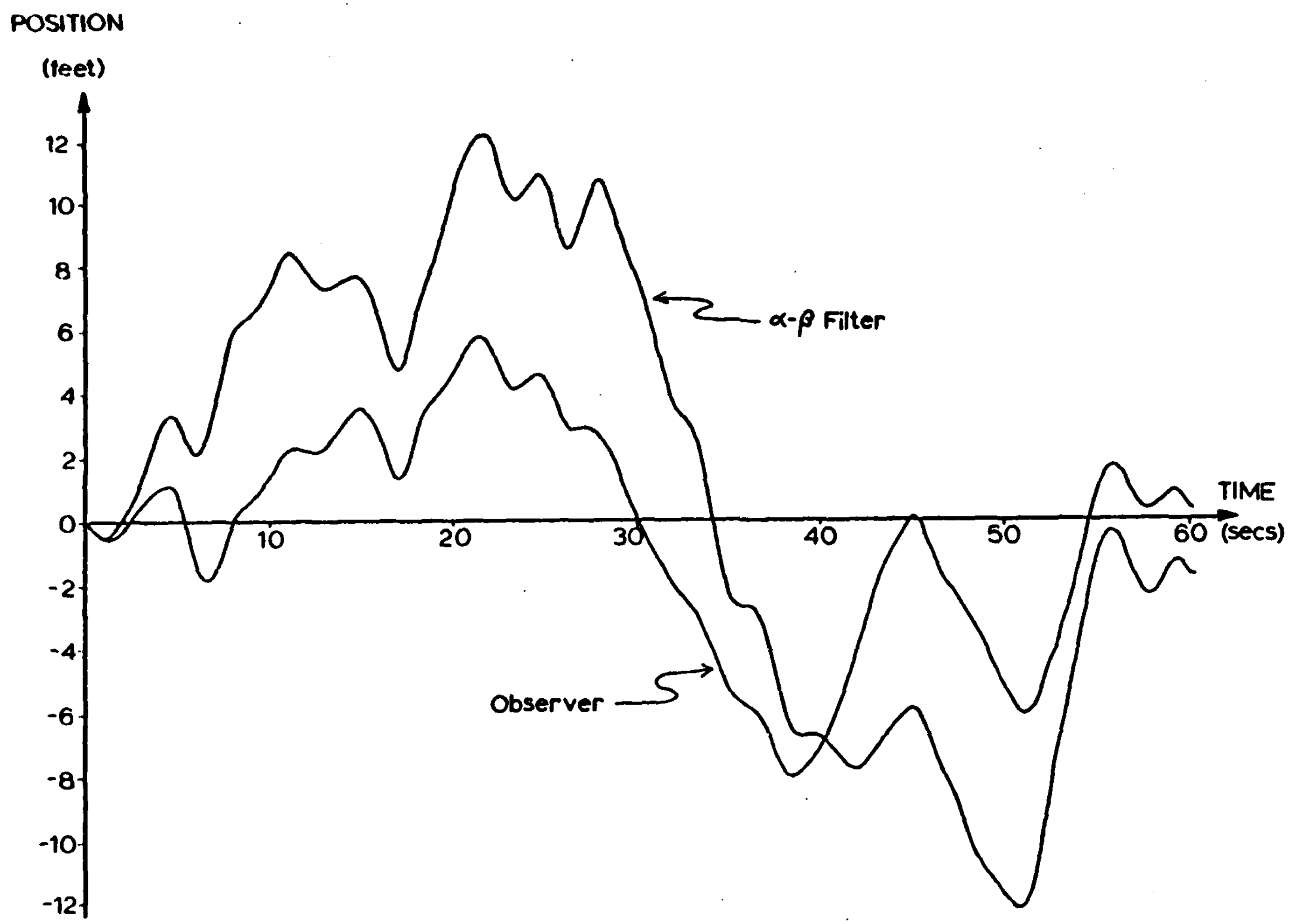


Figure 1-3. Response of the Observer Control System Compared to the Response of the α-β Filter Control System, with the Same Wind and Radar Noise Disturbances.

Table 1-1.

Results of Monte Carlo Runs.

	α-β Filter Control System		Observer Control System		% Improvement
	average	r.m.s.	average	r.m.s.	
Wind and Radar Noise	-0.0030954	5.6082811	-0.2171728	3.9342303	35.1%
Wind Only	-0.0950927	1.9006910	-0.0875373	2.1553383	-12.6%
Radar Noise Only	-0.2798429	2.5217228	-0.0506595	1.6845322	39.6%

It was found that the method introduced was the most robust and accurate of the estimators in noise, due to its unique scan rejection capability. In periods of high signal-to-noise ratio the method had less error than the thresholding methods, and was similar in ability to the first moment estimator. Further, the pulse transmissions required to obtain a desired level of performance was much reduced from the thresholding methods employed in the simulation.

Adaptive Filtering Algorithms

Two approaches to adaptive filtering applicable to the MATCALS system are presented in Part Four. The first approach is based upon adaptively selecting the output from either a fixed parameter α-β or a fixed parameter α-β-γ filter. This selection is determined by an algorithm which incorporates an estimate of the tracking error correlation coefficient. The second approach is based upon an algorithm which automatically adjusts the parameters of an α-β filter to adapt to the dynamics under track.

PART TWO

THE DESIGN OF OBSERVERS FOR THE MATCALS SYSTEM

Prepared for

Georgia Institute of Technology
ATLANTA, GEORGIA

Under

Contract 1-A-2550

by

Electrical Engineering Department
Auburn University
Auburn, Alabama

Prepared by: Robert F. Wilson

Reviewed by: Charles L. Phillips

THE DESIGN OF OBSERVERS

FOR THE MATCALS SYSTEM

ABSTRACT

An observer is designed for a reduced order system that represents the lateral system of the F4J aircraft in an automatic landing configuration. This observer is to be used in the aircraft's lateral control system to estimate its lateral position and lateral velocity. The system currently uses an α-β filter to estimate position and velocity. The observer is designed to replace the α-β filter without significantly changing the characteristics of the system. Results that are obtained from simulations of the F4J aircraft lateral control system indicate that the observer improves the system's response.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

The design of control systems for the automatic landing of aircraft has received considerable attention in recent years. Much of this attention has been directed towards military purposes. One such automatic landing system has been developed for the U.S. Navy and is called the Marine Air Traffic Control and Landing System (MATCALS). During the operation of the MATCALS control system, a considerable amount of noise is produced. This noise is present in such a significant amount that the quality of the control system's performance is greatly degraded. The purpose of this report is to present a method that can be used to improve the MATCALS control system's performance. This method consists of incorporating an observer into the control system.

The method of incorporating an observer into the MATCALS control system is illustrated in this report by employing a simulation of a control system of an individual aircraft. The control system simulation to be used is that of the lateral control system of the F4J aircraft. Once an observer has been incorporated into this control system simulation, results will be presented to show the effect that the observer has on the performance of the control system.

A discussion of observers is given in Chapter II. This discussion includes a definition of observers and how they are designed. A general description of the MATCALS control system and a detailed description of the F4J aircraft lateral control system is presented in Chapter III. This

detailed description is directed towards the simulation of the F4J aircraft lateral control system. The process of designing an observer fcr the lateral control system is given in Chapter IV. Chapter V presents a discussion of the effectiveness of the observer when used in the F4J aircraft lateral control system.

II. OBSERVERS

In optimal control theory, the design of the controlling device is often developed on the assumption that all the states of the system being controlled are in some way available for direct measurement. By knowing the system's states, along with a description of its dynamics, the future behavior of the system can be determined. One way of determining this future behavior is through the use of the system's state equation model (2-1). With all this information available, a scheme can then be developed where an input can be calculated to control the system in the least costly manner. The system model of linear, time-invariant discrete-time system can be expressed as

$$\underline{x}(k+1) = A\underline{x}(k) + B\underline{u}(k) \tag{2-1}$$

where

$\underline{x}(k)$ is an nx1 state vector

$\underline{u}(k)$ is an mx1 input vector

A is an nxn system matrix

B is an nxm input distribution matrix

Associated with this system equation is an output matrix equation

$$\underline{y}(k) = C\underline{x}(k) \tag{2-2}$$

where

$\underline{y}(k)$ is an px1 output vector

C is an pxn output matrix

An unfortunate aspect of modern control theory is that in most realistic systems the total state vector is not available for direct measurement. So either this way of controlling a system is impractical, or a method of evaluating an exceptable estimation of the state vector must be found. This need for a means to estimate the state vector has led to the development of observers. The observer, sometime known as an estimator, was first purposed and developed by Luenberger [1] - [3]. An observer-estimator will be defined as a system that reconstructs the state vector of another system.

Actually, almost any system may be used as an observer-estimator [4]. All that is needed is to use the input and the output of the system that is to be observed as the inputs to the system being used as the observer. Now the state vector of the observer will be some linear transform of the state vector of the original system. However, using this type of observer scheme does not guarantee the quality of the estimated state vector. But, realizing that almost any system can be used as an observer shows the freedom in the design of observers.

Observability

Prior to the designing and the implementation of an observer, it is necessary to determine if the system that is to be observed is in fact observable. If the system in question is described by (2-1) and (2-2) the observability of the system can be confirmed if the following is true [6]:

I. The system of (2-1) and (2-2) is observable if every dynamic mode in the system matrix A is connected to the output vector $\underline{y}(k)$ through the output matrix C.

II. The system of (2-1) and (2-2) is observable, if for any initial value $\underline{x}(0)$, there is a finite N such that $\underline{x}(0)$ can be computed from the observations of $\underline{y}(0)$, $\underline{y}(1)$, ..., $\underline{y}(N-1)$.

An analytical test for observability will be developed according to definition II.

Let the inputs, $\underline{u}(k)$, to the system be zero and set the initial values $\underline{x}(0) = \underline{x}_0$. The system is now described by

$$\underline{x}(k+1) = A\underline{x}(k) \tag{2-3}$$

$$\underline{y}(k) = C\underline{x}(k) \tag{2-4}$$

$$\underline{x}(0) = \underline{x}_0 \tag{2-5}$$

The outputs $\underline{y}(k)$, for k = 0, 1, 2, ..., N are

$$\begin{aligned}
\underline{y}(0) &= C\underline{x}(0) = C\underline{x}_0 \\
\underline{y}(1) &= C\underline{x}(1) = CA\underline{x}(0) = CA\underline{x}_0 \\
\underline{y}(2) &= C\underline{x}(2) = CA\underline{x}(1) = CA^2\underline{x}(0) = CA^2\underline{x}_0 \\
&\vdots \\
\underline{y}(N-1) &= CA^{N-1}\underline{x}_0
\end{aligned}$$

Putting these into matrix form gives

$$\begin{bmatrix} \underline{y}(0) \\ \underline{y}(1) \\ \cdot \\ \cdot \\ \cdot \\ \underline{y}(N-1) \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \cdot \\ \cdot \\ \cdot \\ CA^{N-1} \end{bmatrix} \underline{x}_0 \tag{2-6}$$

To solve for the vector $\underline{x}_0$, it is necessary for the coefficient matrix to be invertable, and to be invertable a matrix must be nonsingular. It is readily apparent that the number of columns of the coefficient matrix is the same as the order of the system, n, and that the number of rows of the matrix is N. Therefore, if N is less than n (2-6) is unsolvable for $\underline{x}_0$, and if N is greater than n, rows CA^n, CA^{n+1}, on up to CA^{N-1} will be added. But, by the Cayley-Hamilton Theorem [7], it can be shown that these new rows will be linear combinations of the lower order rows; therefore, these new rows will not increase the rank of the matrix. Thus, if the system of (2-1) and (2-2) is to be observable, the coefficient matrix of (2-6) must be of rank n. Therefore the test for observability is that the (square) matrix, θ,

$$\theta = \begin{bmatrix} C \\ CA \\ CA^2 \\ \cdot \\ \cdot \\ \cdot \\ CA^{n-1} \end{bmatrix} \tag{2-7}$$

must be nonsingular.

For convenience, the system described by (2-1) and (2-2) will be assumed observable throughout the remainder of this chapter.

Design

A review of the literature indicated that there are two different design procedures of observers for discrete systems. The first of these was developed by Tse and Athans, [4] - [5], and the second procedure was found in a book by Franklin and Powell [6]. The design procedure of both are based on the idea that all available information is to be used; i.e., both the inputs and the outputs of the system to be observed will be used. These input and output signals should be as noiseless as possible for best results. If noise is a significant problem, then the use of a Kalman filter to estimate the state vector should be investigated [6].

Tse-Athans Observer

This observer design was developed for a linear time-varying discrete system. However, since the time-invariant system is just a more restricted case of the time-varying system, the design is applicable to this case.

Before starting the description of this design, the following definitions and conditions must be given.

Let M_{ab} denote the set of axb real valued matrices. The expression $H \varepsilon M_{ab}$ will read: the matrix H is an element of the set of matrixes M_{ab}. If $a \leq b$ then the null space of a matrix $H \varepsilon M_{ab}$ will be denoted by N[H] where

$$N[H] = \{\underline{\alpha};\ H\underline{\alpha} = \underline{0}_a\} \tag{2-8}$$

where $\underline{\alpha}$ is an bx1 vector and $\underline{0}_a$ is the zero ax1 vector.

Now let the matrix $C \varepsilon M_{pn}$ be of rank p; then the set

$$\Omega[C;p,s,n] = \{T \varepsilon M_{sn} : N[T] \cap N[C] = \underline{0}_n\} \tag{2-9}$$

This set is called the set of complimentary matrices of order s for the matrix C if $s \geq n-p$. For a more complete description of these definitions see [4] and the references therein.

Suppose the state equation model of (2-1) and (2-2) is the system to be observed.

$$\underline{x}(k+1) = A\underline{x}(k) + B\underline{u}(k) \tag{2-1}$$

$$\underline{y}(k) = C\underline{x}(k) \tag{2-2}$$

An observer can be designed which has a state equation model of

$$\underline{z}(k+1) = F\underline{z}(k) + G\underline{u}(k) + D\underline{y}(k) \tag{2-10}$$

where

$\underline{z}(k)$ is an sx1 state vector of the observer

$\underline{u}(k)$ is an mx1 input vector of the system

$\underline{y}(k)$ is an px1 output vector of the system

F is an sxs system matrix of the observer

G is an sxm input distribution matrix of the observer

D is an sxp output distribution matrix of the observer

This system is called an s-order observer where s can take on any integer value greater than or equal to n-p.

With the appropriate choice of the initial value of $\underline{z}(0)$ and the sxn matrix T, the following will be true.

$$\underline{z}(k) = T\underline{x}(k) \tag{2-11}$$

where $\underline{z}(0)$ will be a guess and the matrix T will conform to

$$T \in \Omega[C;\ p,\ s,\ n] \tag{2-12}$$

The matrix T fits this condition if and only if there exist an nxs P matrix and an nxp V matrix such that

$$PT + VC = I \tag{2-13}$$

where the matrix I is an nxn identity matrix.

With the chosen T matrix and (2-13) satisfied, the matrices F, D, and G can be evaluated by

$$F = TAP \tag{2-14a}$$

$$D = TAV \tag{2-14b}$$

$$G = TB \tag{2-14c}$$

where the matrices P and V satisfy (2-13) and A and B come from (2-1) of the original system. Substituting (2-14) into (2-10) we get

$$\underline{z}(k+1) = TAP\underline{z}(k) + TB\underline{u}(k) + TAV\underline{y}(k) \tag{2-15}$$

Equation (2-15) can be shown to be an observer of the system described in (2-1) and (2-2) by evaluating

$$\begin{aligned}
&\underline{z}(k+1) - T\underline{x}(k+1) = \\
&TAP\underline{z}(k) + TB\underline{u}(k) + TAV\underline{y}(k) - TA\underline{x}(k) - TB\underline{u}(k) \\
&= TAP\underline{z}(k) + TAV\underline{y}(k) - TA\underline{x}(k) \\
&= TAP\underline{z}(k) + TAVC\underline{x}(k) - TA\underline{x}(k)
\end{aligned}$$

Solving for VC from (2-13),

$$VC = I - PT$$

Then

$$\underline{z}(k+1) - T\underline{x}(k+1) = TAP\underline{z}(k) + TA[I-PT]\underline{x}(k) - TA\underline{x}(k)$$
$$= TAP\underline{z}(k) + TA\underline{x}(k) - TAPT\underline{x}(k) - TA\underline{x}(k)$$
$$= TAP[\underline{z}(k) - T\underline{x}(k)] \qquad (2\text{-}16)$$

Therefore if the initial value $\underline{z}(0)$ is chosen to be $T\underline{x}(0)$, $\underline{z}(k+1)$ is equal to $T\underline{x}(k+1)$ and the observer described by (2-15) will be an observer of the system described by (2-1) and (2-2).

Now that an observer design has been developed, a way to reconstruct the state vector, $\underline{x}(k)$ will be shown. Let the nx1 vector $\hat{\underline{x}}(k)$ be given by

$$\hat{\underline{x}}(k) = P\underline{z}(k) + V\underline{y}(k) \qquad (2\text{-}17)$$

The vector $\hat{\underline{x}}(k)$ can be shown to be an estimate of the state vector $\underline{x}(k)$ by substituting for $\underline{z}(k)$ and $\underline{y}(k)$ with (2-11) and (2-2) respectively. Thus

$$\hat{\underline{x}}(k) = PT\underline{x}(k) + VC\underline{x}(k)$$
$$= [PT + VC]\underline{x}(k)$$

Therefore

$$\hat{\underline{x}}(k) = \underline{x}(k)$$

using (2-13). It is necessary here, as it was for the observer equation in (2-15), that for good results a good choice of $\underline{z}(0)$ is required. That is,

$$\underline{z}(0) = T\underline{x}(0) \qquad (2\text{-}18)$$

In other words, knowledge of the initial values of the originals system states is necessary. If the values chosen for the initial $\underline{z}(0)$ are in

error, then this error will be propagated on through the sequential values of $\underline{z}(k)$. A pictorial representation of (2-15) and (2-17) is shown in Figure 2-1.

Franklin-Powell Observer

The approach taken by Franklin and Powell to observe the state vector of the system

$$\underline{x}(k+1) = A\underline{x}(k) = B\underline{u}(k) \tag{2-1}$$

$$\underline{y}(k) = C\underline{x}(k) \tag{2-2}$$

was, at first, to build a model of the original system and then just measure the readily available state vector of this model. The state equation of this observer is

$$\hat{\underline{x}}(k+1) = A\hat{\underline{x}}(k) + B\underline{u}(k) \tag{2-19}$$

where the vector $\hat{\underline{x}}(k)$ will be the estimate of the state vector $\underline{x}(k)$. This scheme of observing the state vector of the original system should work if the initial values of $\hat{\underline{x}}(0)$ can be set equal to the initial values of $\underline{x}(0)$ and if an accurate system model is available. This "open-loop" observing scheme is shown in Figure 2-2.

However, if the initial value of $\hat{\underline{x}}(0)$ is incorrect then the estimation of the future values of the state vector will also be incorrect. The error of these estimates will be defined as $\tilde{\underline{x}}(k)$, where

$$\tilde{\underline{x}}(k) = \underline{x}(k) - \hat{\underline{x}}(k) \tag{2-20}$$

Then the error's difference equation is

$$\begin{aligned} \underline{x}(k+1) - \hat{\underline{x}}(k+1) &= A\underline{x}(k) + B\underline{u}(k) - A\hat{\underline{x}}(k) - B\underline{u}(k) \\ \tilde{\underline{x}}(k+1) &= A\tilde{\underline{x}}(k) \end{aligned} \tag{2-21}$$

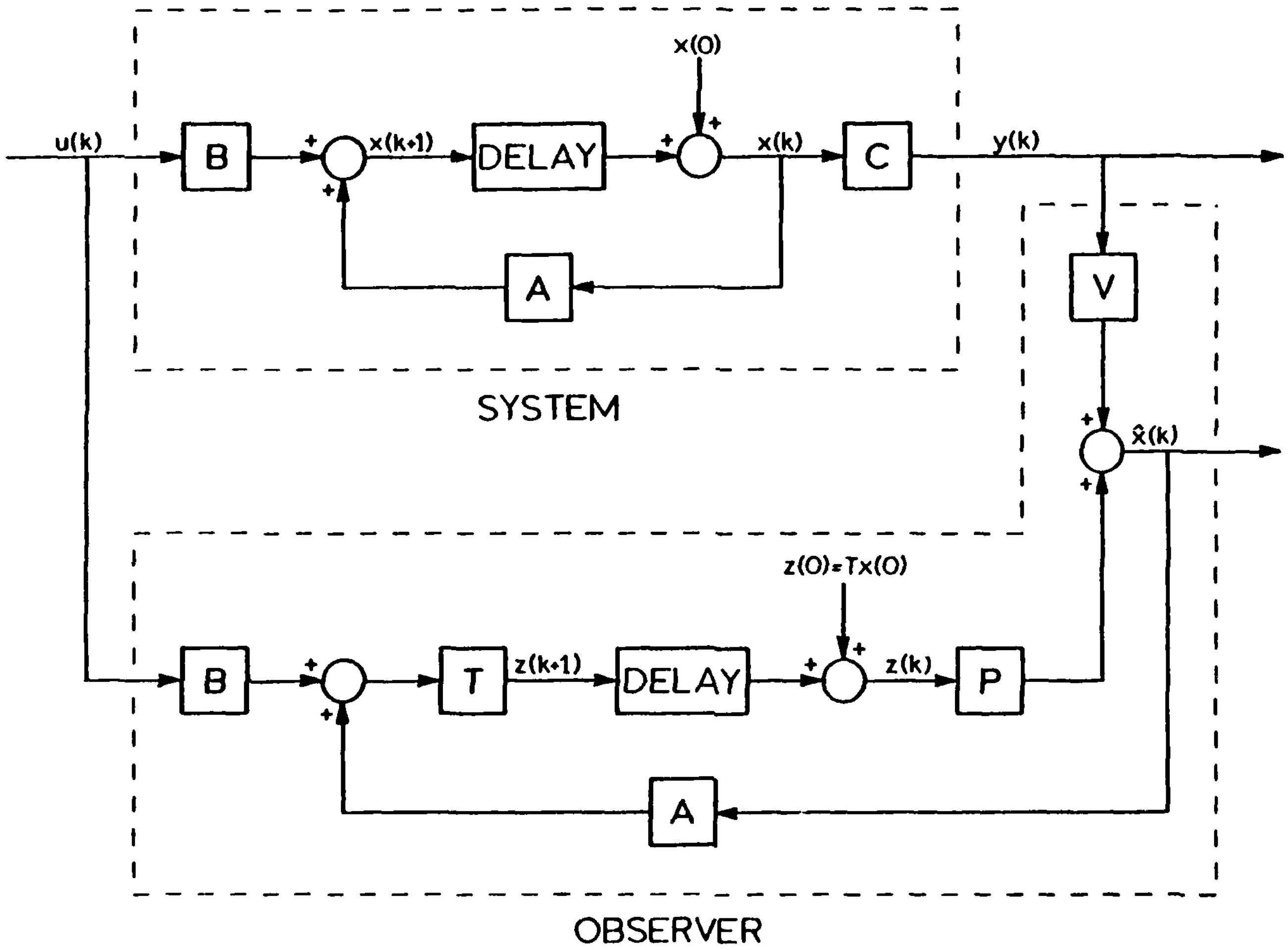


Figure 2-1. Structure of Observer. Vector $\hat{\underline{x}}(k)$ Estimates the State Vector $\underline{x}(k)$.

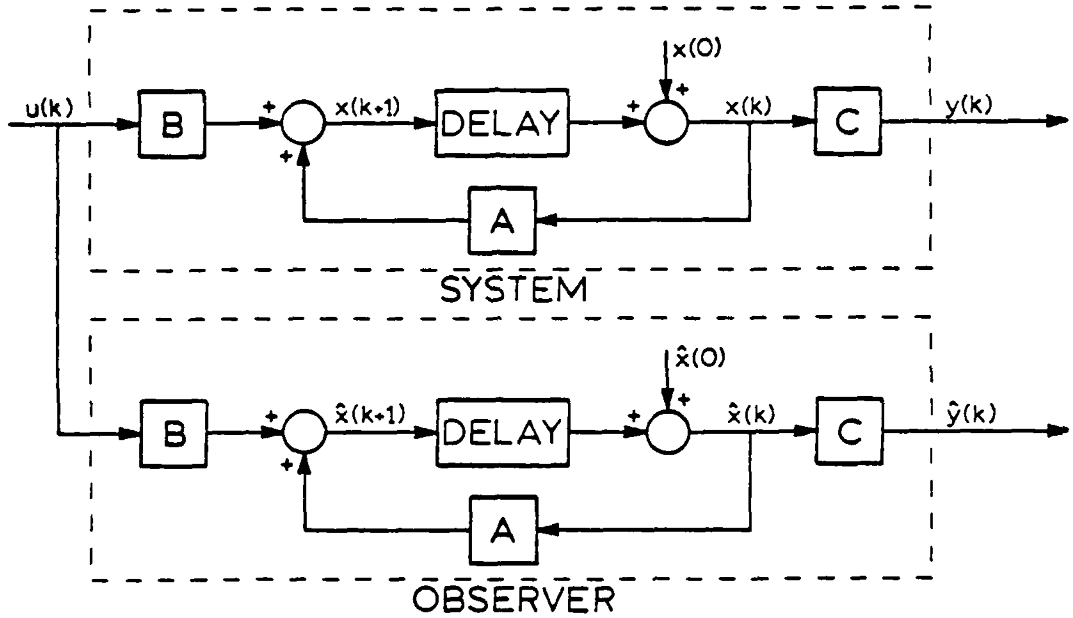


Figure 2-2. "Open-Loop" Observer.

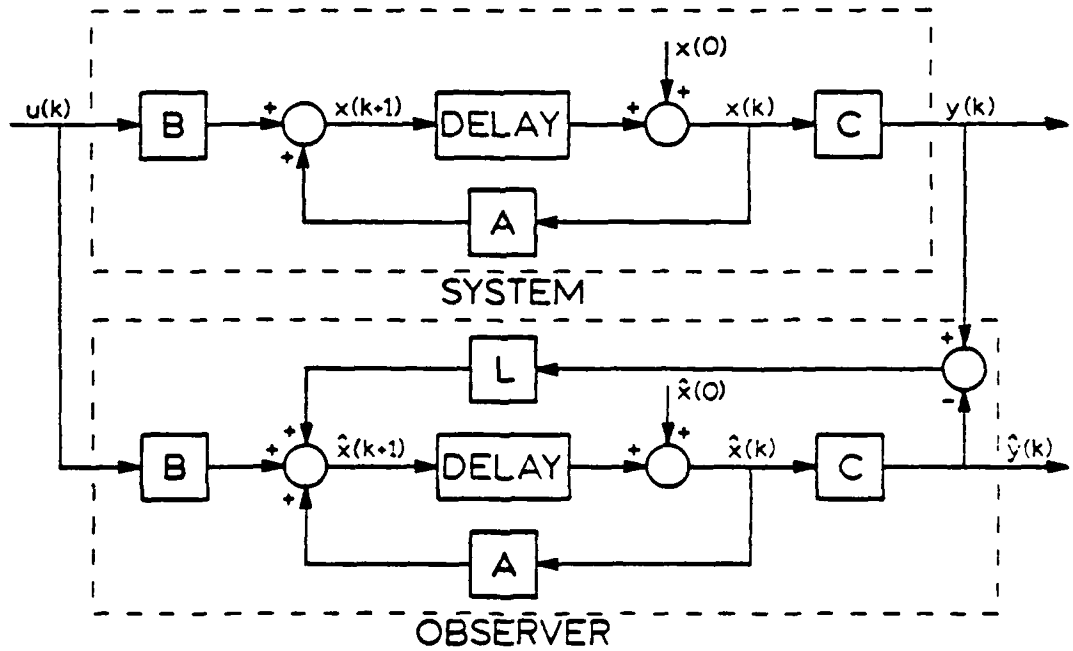


Figure 2-3. "Closed-Loop" Observer.

As can be seen, the error's dynamics are the same as the original system's dynamics. Thus if the system is always in motion then the error will also be in motion with the same dynamics; therefore the error will not disappear.

A way to compensate for this error is to feed back the difference between the measured output of the original system and the corresponding output of the observer, as is shown in Figure 2-3. The feeding back of this difference signal will constantly correct the observer, thereby minimizing the error $\tilde{\underline{x}}(k)$. The state equation of this scheme is

$$\hat{\underline{x}}(k+1) = A\hat{\underline{x}}(k) + B\underline{u}(k) + L[\underline{y}(k) - C\hat{\underline{x}}(k)] \tag{2-22}$$

where the gain matrix L will be nxp. Gathering terms will give the following state equation

$$\hat{\underline{x}}(k+1) = [A-LC]\hat{\underline{x}}(k) + B\underline{u}(k) + Ly(k) \tag{2-23}$$

Again a difference equation is found for the error

$$\begin{aligned}\underline{x}(k+1) - \hat{\underline{x}}(k+1) &= A\underline{x}(k) + B\underline{u}(k) - A\hat{\underline{x}}(k) - B\underline{u}(k)\\ &\quad - L\underline{y}(k) + LC\hat{\underline{x}}(k)\end{aligned}$$

Substituting for $\underline{y}(k)$ with (2-2) yields

$$\tilde{\underline{x}}(k+1) = A\underline{x}(k) - A\hat{\underline{x}}(k) - LC\underline{x}(k) + LC\hat{\underline{x}}(k)$$

$$\tilde{\underline{x}}(k+1) = [A-LC]\tilde{\underline{x}}(k) \tag{2-24}$$

Now the error dynamics are seen to be determined by the matrix [A-LC], and with a proper choice of the matrix L, the error's dynamics and thus the observer's dynamics, can be made "faster", thereby causing $\tilde{\underline{x}}(k)$ to converge to zero in a more satisfactory manner than in the "open-loop" observer. To say this another way, the vector $\hat{\underline{x}}(k)$ will converge to the

state vector $\underline{x}(k)$ faster regardless of the value of $\hat{\underline{x}}(0)$, if a good choice of the gain matrix L is made.

Another advantage of this "closed-loop" observer over the "open-loop" observer is that if the matrices A and B of the observer are not exactly the same as the matrices A and B of the system, the error caused by these inaccuracies are made acceptably small.

The determination of the gain matrix L can be done in two different ways. The first way is by matching coefficients and the second is to use Ackermann's estimation formula. Both ways assume that the desired pole locations of the observer are known.

Matching Coefficients. The matching coefficient technique of calculating the gain matrix L is the "brute force" method. First it is necessary to expand the determinant

$$\alpha_L(z) = |zI - [A-LC]| \tag{2-25}$$

which will give the characteristic polynomial of the observer in terms of the elements of the matrix L. Next, the desired observer characteristic polynomial is expanded.

$$\alpha(z) = (z-P_1)(z-P_2) \ldots (z-P_n) \tag{2-26}$$

where the P_i's are the desired pole location of the observer. All that is needed now is to set $\alpha_L(z)$ of (2-25) equal to $\alpha(z)$ of (2-26) and solve for the elements of the gain matrix L.

Ackermann's Formula. For a single-output system, a more systematic method of computing the gain matrix L is through the use of Ackermann's Formula

$$L = \alpha(A) \begin{bmatrix} C \\ CA \\ CA^2 \\ \cdot \\ \cdot \\ \cdot \\ CA^{n-1} \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \\ 1 \end{bmatrix} \tag{2-27}$$

The polynomial $\alpha(A)$ is the observer's desired characteristic polynomial described in (2-26), with the complex variable z replaced by the system matrix A. The coefficient matric that has the rows of C, CA, CA^2, ... CA^{n-1}, is recognized to be the observability matrix, θ, described in (2-7). This matrix is square, since the C is a row matrix. Finally the vector of zeroes and one 1 is an nx1 unit vector. The development of Ackermann's Formula is given in Appendix A, and a BASIC program to compute the gain matrix L, based on Ackermann's Formula, is given in Appendix B.

Reduced Order Observer

The reconstruction of the entire state vector of a system is not necessary, when some of the system states are directly measurable. Therefore, it is not necessary for the observer to have the same order as the system. The minimum order that an observer can have is no less than n-p, where n is the order of the system being observed and p is the rank of the output matrix. In other words there is no need to reconstruct states that are already available from the output of the original system. But, if there is significant noise on the measurements of the system, better results are obtained with the use of a full order observer.

Again, take the system described by (2-1) and (2-2)

$$\underline{x}(k+1) = A\underline{x}(k) + B\underline{u}(k) \qquad (2\text{-}1)$$

$$\underline{y}(k) = C\underline{x}(k) \qquad (2\text{-}2)$$

The state vector, $\underline{x}(k)$, can be divided into two parts. The first contains the states that are measured, $\underline{x}_a(k)$, and the second contains the states that are not measured, $\underline{x}_b(k)$. This division gives the following partitioned system state equations

$$\begin{bmatrix} \underline{x}_a(k+1) \\ \hline \underline{x}_b(k+1) \end{bmatrix} = \left[\begin{array}{c|c} A_{aa} & A_{ab} \\ \hline A_{ba} & A_{bb} \end{array}\right] \begin{bmatrix} \underline{x}_a(k) \\ \hline \underline{x}_b(k) \end{bmatrix} + \begin{bmatrix} B_a \\ \hline B_b \end{bmatrix} \underline{u}(k) \qquad (2\text{-}28)$$

$$\underline{y}(k) = \begin{bmatrix} I & 0 \end{bmatrix} \begin{bmatrix} \underline{x}_a(k) \\ \hline \underline{x}_b(k) \end{bmatrix}$$

Solving for the measured and unmeasured states gives

$$\underline{x}_a(k+1) = A_{aa}\underline{x}_a(k) + A_{ab}\underline{x}_b(k) + B_a\underline{u}(k) \qquad (2\text{-}29)$$

$$\underline{x}_b(k+1) = A_{ba}\underline{x}_a(k) + A_{bb}\underline{x}_b(k) + B_b\underline{u}(k) \qquad (2\text{-}30)$$

Equation (2-30) can be treated as a system's state equation with the (n-p)x1 vector $\underline{x}_b(k)$ as the state vector and the (n-p)x(n-p) matrix A_{bb} as the system matrix. The other two term, $A_{ba}\underline{x}_a(k)$ and $B_b\underline{u}(k)$ are known, so therefore they can be treated as the input. With equation (2-30) treated as just described, equation (2-29) can be used as the output matrix equation, where the matrix A_{ab} will be the px(n-p) output matrix and the output is seen to be equal to $\underline{x}_a(k+1) - A_{aa}\underline{x}_a(k) - B_a\underline{u}(k)$.

Summarizing the above substitutions

$$\underline{x}(k) \leftarrow \underline{x}_b(k)$$

$$A \leftarrow A_{bb}$$

$$B\underline{u}(k) \leftarrow A_{ba}\underline{x}_a(k) + B_b\underline{u}(k)$$

$$\underline{y}(k) \leftarrow \underline{x}_a(k+1) - A_{aa}\underline{x}_a(k) - B_a\underline{u}(k)$$

$$C \leftarrow A_{ab}$$

and using these in the observer equation (2-22) will give the following reduced-order observer equation

$$\hat{\underline{x}}_b(k+1) = A_{bb}\hat{\underline{x}}_b(k) + A_{ba}\underline{x}_a(k) + B_b\underline{u}(k)$$

$$+ L[\underline{x}_a(k+1) - A_{aa}\underline{x}_a(k) - B_a\underline{u}(k) - A_{ab}\hat{\underline{x}}_b(k)] \qquad (2\text{-}31)$$

Gathering terms and rewriting gives

$$\hat{\underline{x}}_b(k+1) = [A_{bb}-L\ A_{ab}]\hat{\underline{x}}_b(k) + [A_{ba}-L\ A_{aa}]\underline{x}_a(k)$$

$$+ [B_b-L\ B_a]\underline{u}(k) + L\ \underline{x}_a(k+1) \qquad (2\text{-}32)$$

The error state equation can be derived by subtracting (2-31) from (2-30)

$$\underline{x}_b(k+1) - \hat{\underline{x}}_b(k+1)$$

$$= A_{ba}\underline{x}_a(k) + A_{bb}\underline{x}_b(k) + B_b\underline{u}(k)$$

$$-A_{bb}\hat{\underline{x}}_b(k) - A_{ba}\underline{x}_a(k) - B_b\underline{u}(k)$$

$$- L[\underline{x}_a(k+1) - A_{aa}\underline{x}_a(k) - B_a\underline{u}(k) - A_{ab}\hat{\underline{x}}_b(k)$$

$$= A_{bb}\underline{x}_b(k) - A_{bb}\hat{\underline{x}}_b(k) - L[\underline{x}_a(k+1) - A_{aa}\underline{x}_a(k) - B_a\underline{u}(k)$$

$$- A_{ab}\hat{\underline{x}}_b(k)]$$

Substituting (2-29) in for $\underline{x}_a(k+1)$ gives

$$\tilde{\underline{x}}_b(k+1) = A_{bb}\underline{x}_b(k) - A_{bb}\hat{\underline{x}}(k) - L[A_{ab}\underline{x}_b(k) - A_{ab}\hat{\underline{x}}(k)]$$

$$= A_{bb}\tilde{\underline{x}}_b(k) - L\ A_{ab}\tilde{\underline{x}}_b(k)$$

$$= [A_{bb} - L\ A_{ab}]\tilde{\underline{x}}_b(k) \qquad (2\text{-}33)$$

As can be seen, the matrix L can be used again to make the error vector $\tilde{x}(k)$, converge to zero relatively "fast". The gain matrix L, can be calculated as described above if $L\ A_{ab}$ is substituted for LC in (2-25) or by substituting A_{ab} for C and A_{bb} for A in (2-27).

Unknown Inputs

In the observer design procedures presented in this chapter, it has been assumed that all the inputs to the system to be observed are known. But in the real world there are a large percentage of systems that have unknown inputs, such as noise. Therefore, a method to observe the state vector of a system must be found where the need to know all inputs is eliminated. Otherwise the observer can be designed just for the known inputs and the unknown inputs are ignored. If the latter scheme is used, it is hoped that the resulting error will be small.

A way to modify the observer design in such a way that the unknown inputs are not required was purposed by Wang, Davison, and Dorato [8]. This design modification can be used with the Franklin-Powell reduced order observer design. This is true, since in the observer state equation, (2-32) the input distribution matrix is calculated using the variable gain matrix L. This calculation is seen to be $[B_b - L\ B_a]$. Now, for example, assume that the system to be observed has two inputs, such that $u_1(k)$ is known and $u_2(k)$ is unknown. The idea now is to find a matrix L that will make the second column of the matrix $[B_b - L\ B_a]$ vanish. In this way no matter what the unknown input $u_2(k)$ does, the observer will operate properly. It should be noted that if this

modification is used, the freedom of picking the pole locations of the observer is forfeited. Therefore, the L matrix found by this method should be examined, because the poles that correspond to this calculated L matrix might not be satisfactory. In fact this L matrix could make the observer an unstable system.

Comment on Pole Locations

The process of picking the observer's pole locations is restricted by only one rule; that is, the chosen pole locations should not cause the observer to be unstable. In practice, the observer's roots are picked so that the observer will be somewhat faster than the system being observed. The upper limit to the observer's speed is restricted by how much noise there is on the measurements and by how well the system has been modeled, e.g., have any inputs been ignored. This limit can be determined by simulation, or perhaps some optimizing technique could be developed. Very little information has been found in the literature to aid in the choice of the pole locations.

III. SYSTEM DESCRIPTION AND SIMULATION

The system that is to be studied in this paper is, in general, the MATCALS control system. This control system is used in the automatic landing of aircraft. The system consists of three basic parts: the aircraft itself, the radar unit, and the controlling unit. During the operation of this control system, the radar unit measures the approximate vertical and lateral positions of the aircraft, which are then transmitted to the land-based controlling unit. From these approximations the controlling unit calculates appropriate bank and pitch commands. These commands are then transmitted to the aircraft autopilots, which in turn cause the aircraft to respond accordingly. A diagram showing this operation of the MATCALS control system is given in Figure 3-1. A detailed discussion of the MATCALS control system is given in Reference [10].

To facilitate the study of the MATCALS control system, FORTRAN IV programs have been developed at Auburn University to simulate the system. Two simulation programs are available: one for the F4J aircraft control system, and one for the AE7 aircraft control system [9]. In both of these simulations, the control system is divided into two uncoupled subsystems, i.e., a lateral control system and a vertical control system. The study that is to be carried out in this paper will be accomplished through the use of one of these control system simulation programs. Of the four programs available, the program for the lateral control system of the F4J

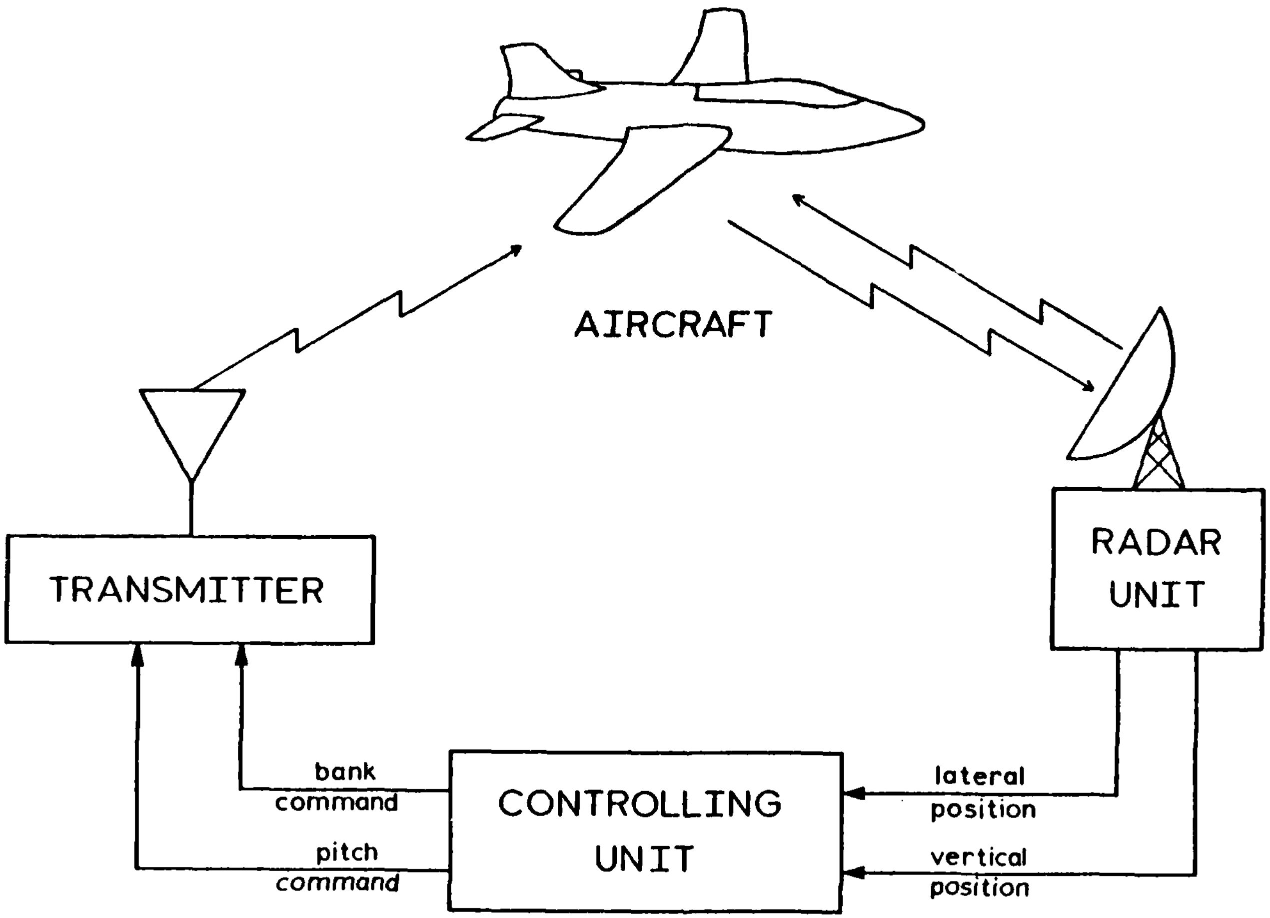


Figure 3-1. Structure of the MATCALS Control System.

aircraft was selected. A description of this lateral control system and its simulation program will now be given. A listing of the simulation program is given in the appendix.

The F4J Aircraft's Lateral Control System

A general description of the lateral control system of the F4J aircraft and the simulation of this system is obtained from the block diagram given in Figure 3-2. From this diagram, it is seen that the control system is modeled as a sampled-data system containing a digital filter. In this model, the continuous part of the system is the aircraft lateral system, and the digital filter is the controlling unit. The sampling effect of this system is modeled in the radar unit. Even though this control system and its simulation program are constructed to be able to operate at various sampling rates, the work presented in this paper will be accomplished with the sampling period T set to 0.1 seconds. A more involved description of this lateral control system, and how it is simulated, will now be provided through a discussion of the three parts of the block diagram given in Figure 3-2.

F4J Aircraft's Lateral System

The dynamics of the F4J aircraft's lateral airframe are described by a sixth order linear differential equation. This differential equation is presented in Reference [9]. As is shown in this reference, the six states of this differential equation represent six physical variables of the aircraft's lateral airframe. These variables are listed below along with their respective symbols.

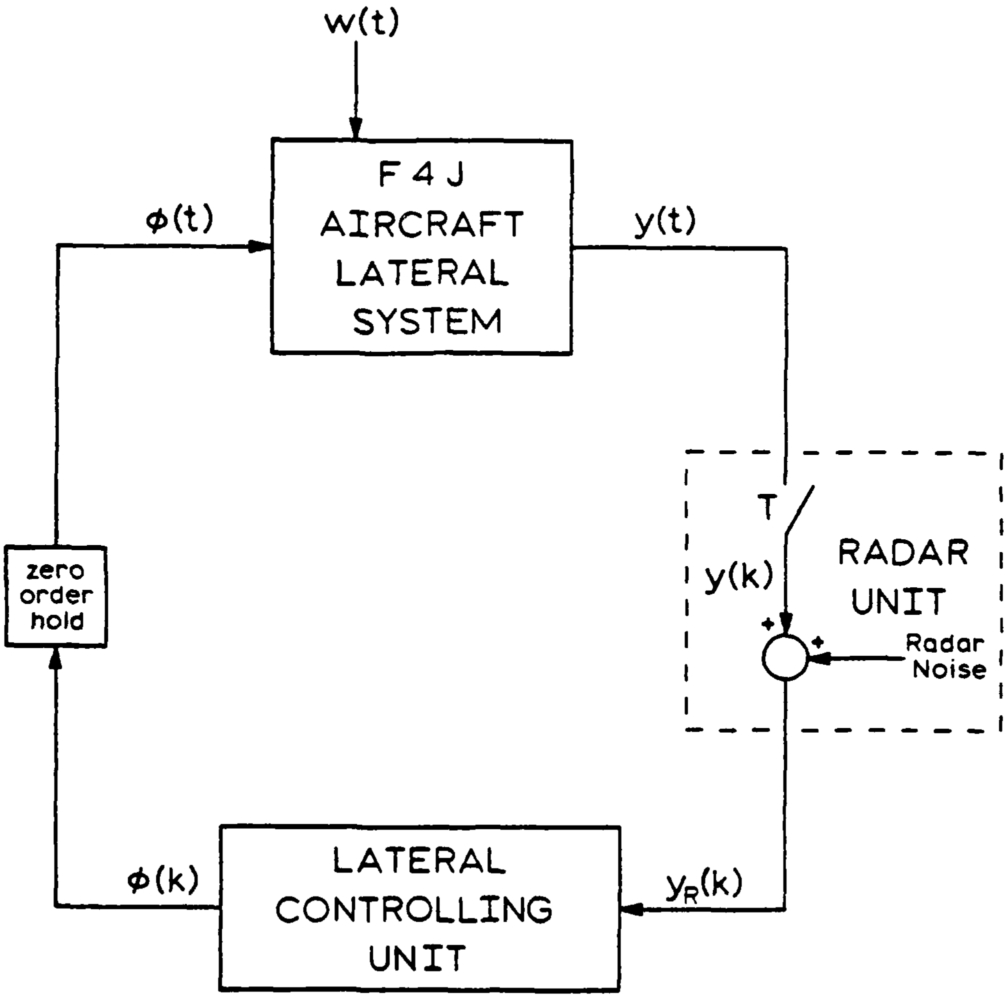


Figure 3-2. Block Diagram of F4J Aircraft Lateral Control System.

$\Delta\delta(t)$ - roll angle perturbation

$\Delta\psi(t)$ - yaw angle perturbation

$\Delta\beta(t)$ - perturbation in the angle of side slip

$\Delta p(t)$ - perturbation in the angle of the x-axis

$\Delta r(t)$ - perturbation in the angle of the z-axis

$q(t)$ - lateral distance from the extended centerline of the runway

To complete the description of the F4J aircraft lateral dynamics, the autopilot dynamics must be combined with the dynamics of the aircraft's lateral airframe. The autopilot dynamics are described by a third order nonlinear differential equation [9]. The three nonlinearities of the autopilot are of the limiter type. Therefore, the complete description of the F4J aircraft's lateral dynamics is given by a ninth order nonlinear differential equation. This description is expressed in a continuous state matrix equation of the form

$$\dot{\underline{x}}(t) = A\underline{x}(t) + B\underline{u}(t) + E\underline{f}(t) \tag{3-1}$$

where

$\underline{x}(t)$ is the 9x1 state vector

$\underline{u}(t)$ is the 2x1 input vector

$\underline{f}(t)$ is the 3x1 nonlinearity vector

A is the 9x9 system matrix

B is the 9x2 input distribution matrix

E is the 9x3 nonlinearity distribution matrix

These vectors and matrices will now be de..ribed as they are found in the simulation.

The state vector. As is mentioned above, six of the nine states of the state vector $\underline{x}(t)$ represent physical variables of the aircraft's lateral airframe. The remaining three states are contributed by the autopilot and represent no physical variables. The following vector gives the assignment of the states in the simulation program.

$$\underline{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \\ x_5(t) \\ x_6(t) \\ x_7(t) \\ x_8(t) \\ x_9(t) \end{bmatrix} = \begin{bmatrix} \left.\begin{matrix} \Delta\delta(t) \\ \Delta\psi(t) \\ \Delta\beta(t) \\ \Delta p(t) \\ \Delta r(t) \\ q(t) \end{matrix}\right\} \text{Airframe States} \\ \left.\begin{matrix} x_7(t) \\ x_8(t) \\ x_9(t) \end{matrix}\right\} \text{Autopilot States} \end{bmatrix} \tag{3-2}$$

The input vector. There are two inputs to the F4J aircraft lateral system. These inputs are the bank command input $\phi(t)$ and the wind input $w(t)$. The positions of these two inputs in the input vector $\underline{u}(t)$ are

$$\underline{u}(t) = \begin{bmatrix} \phi(t) \\ w(t) \end{bmatrix} \tag{3-3}$$

The bank command input is produced by the controlling unit; therefore it is known. The wind input, on the other hand, is not known. This input is modeled in the control system simulation program by a random number generator.

The nonlinearity vector. The vector $\underline{f}(t)$ describes the nonlinearities that are contained in the aircraft lateral autopilot. This vector is

$$\underline{f}(t) = \begin{bmatrix} f_1(t) \\ f_2(t) \\ f_3(t) \end{bmatrix} \tag{3-4}$$

The three nonlinear functions $f_1(t)$, $f_2(t)$, and $f_3(t)$ are given in Table 3-1.

The matrices. The matrices A, B, and E of the state equation, (3-1), that describes the F4J aircraft lateral dynamics are presented in Table 3-2.

To model the continuous F4J aircraft lateral system in the simulation program, an integration algorithm is required. This algorithm is the fourth-order Runge-Kutter integration procedure. This procedure is described in Reference [11], and is given in the simulation listing in Appendix D.

Radar Unit

The radar unit used in the lateral control system of the F4J aircraft is the AN/TPN-22 phased array radar [12]. The purpose of this radar unit is to periodically determine the aircraft's lateral position. Unfortunately, in the process of determining the lateral position, a significant amount of noise is produced. The combination of this noise and the aircraft's sampled lateral position forms the radar output signal.

The description of the simulation of this radar unit will now be presented. This description will be given in three steps. First it will be shown how the lateral position of the aircraft is obtained from the simulation of the aircraft lateral dynamics. Next the effects of sampling

Table 3-1.

Nonlinear Functions.

$$f_1(t) = \begin{cases} 14.0 & \text{if} \quad \phi(t) > 14.0 \\ \phi(t) & \text{if} \quad |\phi(t)| \le 14.0 \\ -14.0 & \text{if} \quad \phi(t) < -14.0 \end{cases}$$

let

$$n(t) = 171.9*x_1(t) + 68.76*x_4(t) - 3.0*f_1(t)$$

$$f_2(t) = \begin{cases} 7.5 & \text{if} \quad n(t) > 7.5 \\ n(t) & \text{if} \quad |n(t)| \le 7.5 \\ -7.5 & \text{if} \quad n(t) < -7.5 \end{cases}$$

let

$$\begin{aligned} m(t) = {} & 80.4631*x_1(t) - 18.0533*x_3(t) + 53.7988*x_4(t) \\ & + 150.2088*x_5(t) - 0.0173*x_7(t) - 1.25*x_8(t) \\ & + 1.268*x_9(t) - 18.0533*w(t) + 0.67*f_2(t) \end{aligned}$$

$$f_3(t) = \begin{cases} 5.0 & \text{if} \quad m(t) > 5.0 \\ m(t) & \text{if} \quad |m(t)| \le 5.0 \\ -5.0 & \text{if} \quad m(t) < -5.0 \end{cases}$$

Table 3-2.

System Matrices.

A Matrix

$$
\begin{bmatrix}
0.0 & 0.0 & 0.0 & 1.0 & 0.1637 & 0.0 & 0.0 & 0.0 & 0.0 \\
0.0 & 0.0 & 0.0 & 0.0 & 1.013 & 0.0 & 0.0 & 0.0 & 0.0 \\
0.1439 & 0.0 & -0.08517 & 0.2361 & -0.9669 & 0.0 & -0.00008179 & 0.0 & 0.005982 \\
0.0 & 0.0 & -6.738 & -1.285 & 0.7475 & 0.0 & -0.05358 & 0.0 & -0.1402 \\
0.0 & 0.0 & 0.6238 & -0.1996 & -0.1114 & 0.0 & -0.008548 & 0.0 & -0.2203 \\
-48.92 & 220.4 & 220.8 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\
0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & -10.0 & 0.0 & 0.0 \\
0.0 & 0.0 & 0.0 & 1.501 & 57.28 & 0.0 & 0.0 & -0.5 & 0.0 \\
0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & -20.0
\end{bmatrix}
$$

Table 3-2.
(cont)

B Matrix

$$\begin{bmatrix} 0.0 & 0.0 \\ 0.0 & 0.0 \\ 0.0 & -0.08517 \\ 0.0 & -0.6738 \\ 0.0 & 0.6238 \\ 0.0 & 0.0 \\ 0.0 & 0.0 \\ 0.0 & 0.0 \\ 0.0 & 0.0 \end{bmatrix}$$

E Matrix

$$\begin{bmatrix} 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 10.0 & 0.0 \\ 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$$

will be added. Finally, an illustration will be given showing how the radar noise is produced and combined with the sampled lateral position signal.

The lateral position is obtained from the state equation description of the F4J aircraft lateral dynamics through the use of the output matrix equation

$$y(t) = C\underline{x}(t) \tag{3-5}$$

This matrix equation will give the desired output if all the elements of the 1x9 output matrix C are zero except the (1, 6) element, which must be unity. This is seen by examining the state vector $\underline{x}(t)$, in (3-2).

The effect of sampling the aircraft lateral position is simulated by testing the signal y(t) of (3-5) every sampling period. As is seen in the block diagram of Figure 3-2 this sampled lateral position signal is the digital signal y(k). It is this signal that is corrupted by the radar noise.

The simulation of the AN/TPN-22 radar noise is illustrated in Figure 3-3 [12]. As is shown in this illustration, the noise is added to the aircraft's lateral angle. This lateral angle is the angle between the centerline of the runway and the projected line between the aircraft and the touchdown point. The lateral angle ta(k) is calculated by

$$ta(k) = \tan^{-1} \frac{y(k) + 178.1}{r(k) + 762.8} \tag{3-6}$$

The variable r(k) is the range of the aircraft from the touchdown point; this is assumed to be known. The values 178.1 and 762.8 are the lateral position and range of the radar unit in respect to the touchdown point.

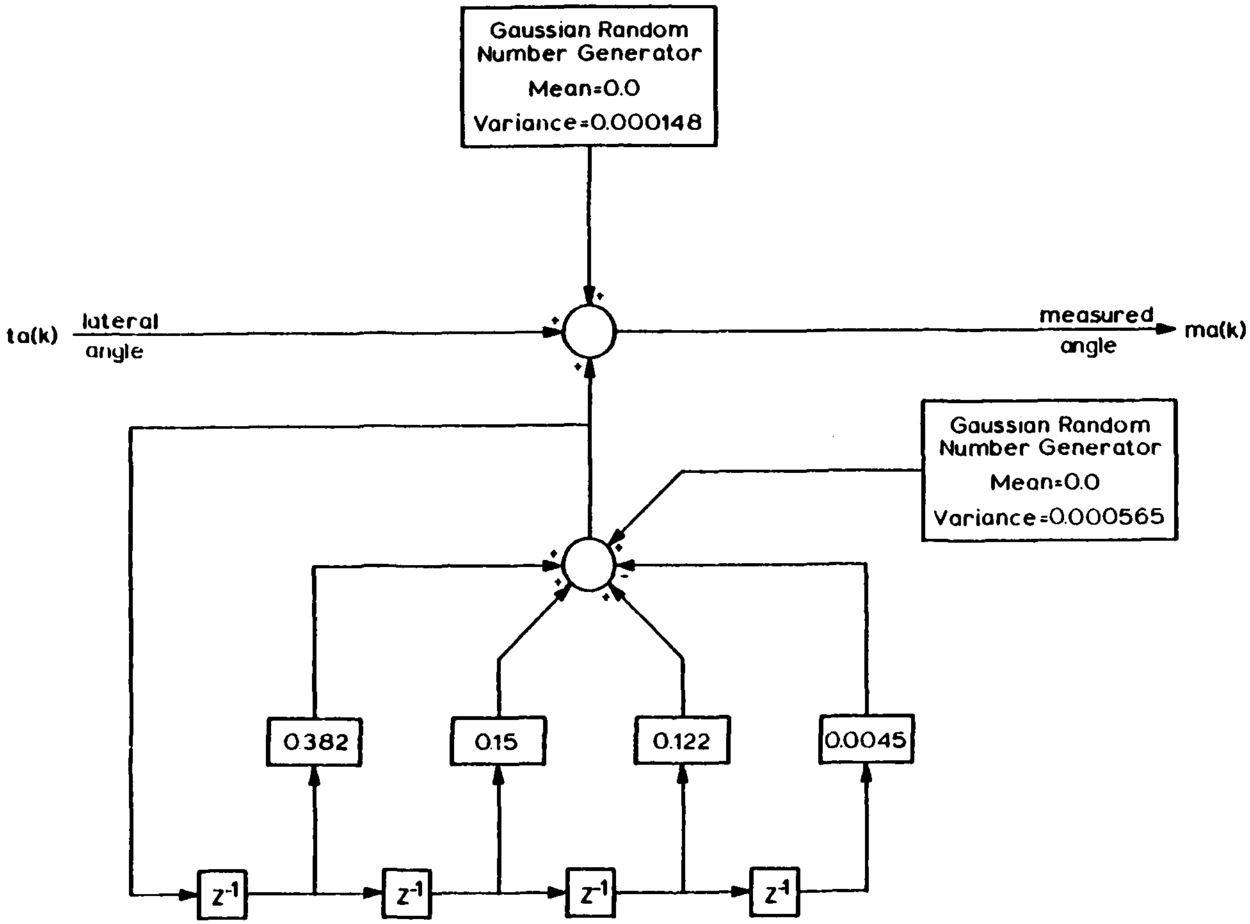


Figure 3-3. Block Diagram of the AN/TPN-22 Radar Noise Simulation.

The noise is then added to this angle ta(k) and the measured angle ma(k) is obtained. The measured angle is transformed into the radar unit's output signal $y_R(k)$ by

$$y_R(k) = [r(k) + 762.9] \tan [ma(k)] - 178.1 \tag{3-7}$$

Controlling Unit

The controlling unit of the aircraft lateral control system is the SPN-42 digital controller. This controller is basically a digital PID (proportional plus integral plus derivative) type controller. The development of the SPN-42 digital controller is described in detail in Reference [12]. The basic form of the controlling unit is given in the block diagram of Figure 3-4. The four α filters are first-order low-pass digital filters used to reduce the effects of high frequency noise. The tracking α-β filter is a second-order digital filter used to determine estimates of the aircraft's lateral position and lateral velocity from the noisy lateral position radar signal. These estimates, $\hat{y}(k)$ and $\hat{\dot{y}}(k)$, are then passed on to the remainder of the controlling unit where the bank command $\phi(k)$ is calculated.

The simulation of the controlling unit is obtained from the signal flow graph of the SPN-42 digital controller given in Figure 3-5 [12]. This flow graph includes five nonlinearities that are present in the filter, but were omitted in the block diagram of Figure 3-4. Four of these nonlinearities are of the limiter type and the fifth is a floating limiter. A floating limiter is a discrete nonlinearity that limits the amount of change that can occur in one sampling period.

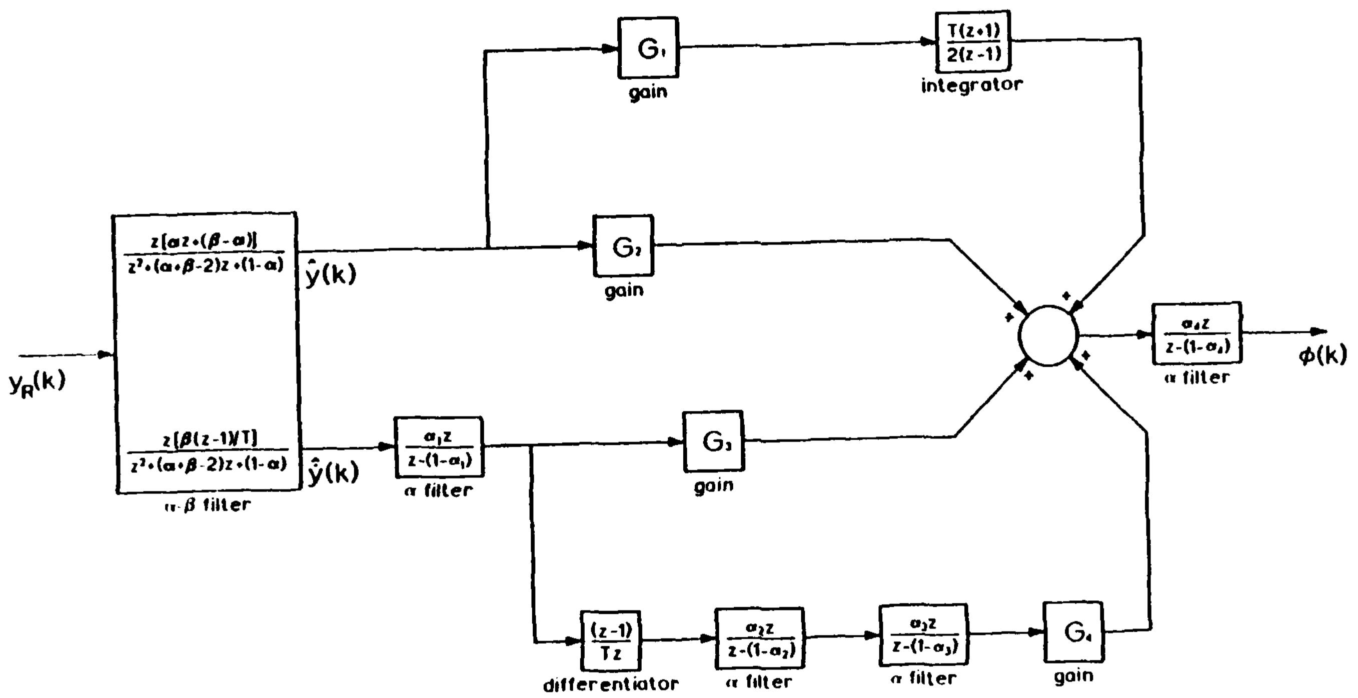


Figure 3-4. Block Diagram of Linearized SPN-42 Digital Controller with Transfer Functions.

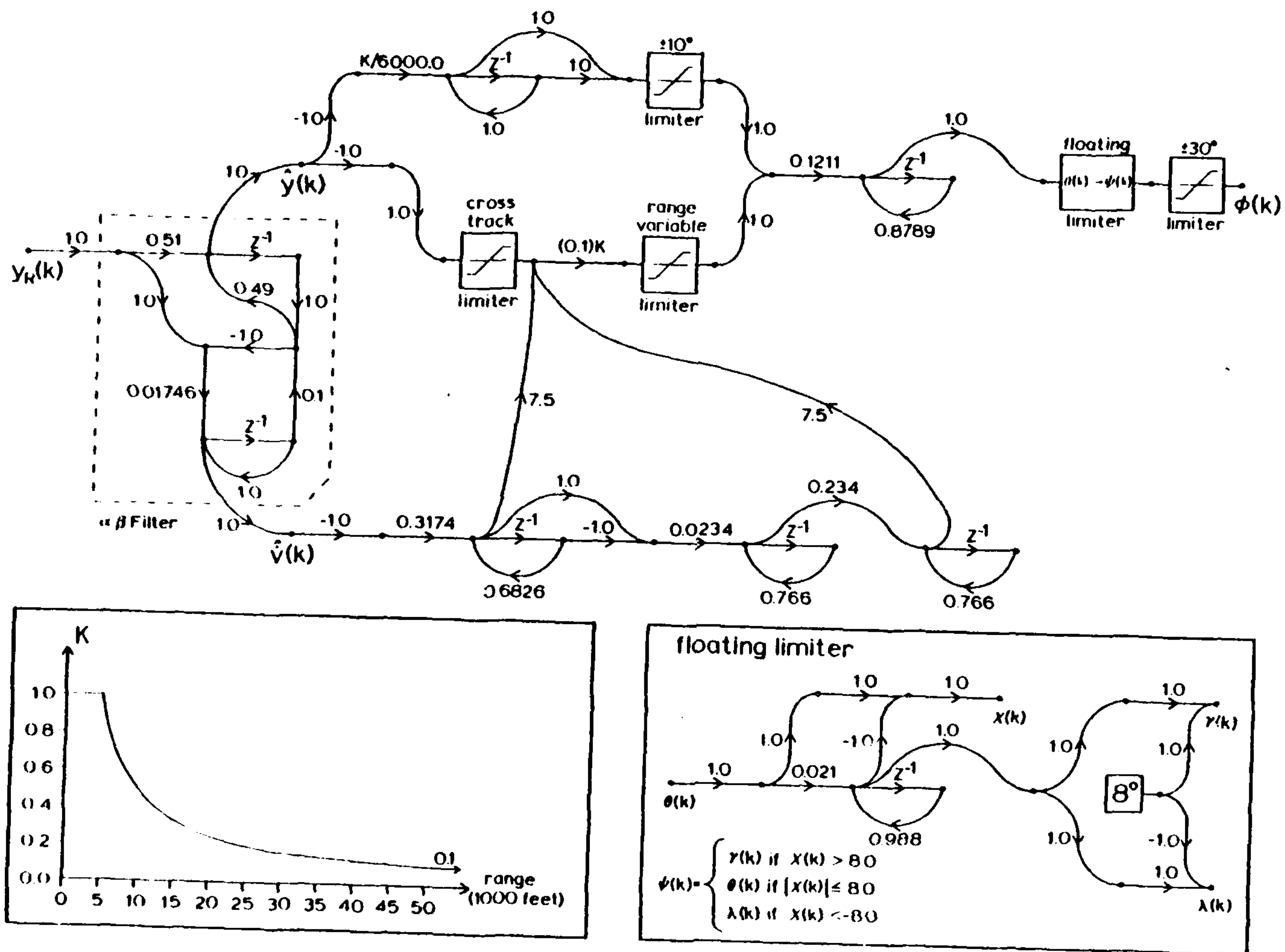


Figure 3-5. Signal Flow Graph of the SPN-42 Digital Controller.

Since the controlling unit of the lateral control system is a digital device, it processes information only at given instants of time. These instants are synchronized with the sampler modeled in the radar unit and have a time duration that is very short with respect to the length of the sampling period. For this reason, the process of sampling the aircraft lateral position to the outputting of the bank command is modeled with no time delay in the simulation.

As is shown in Figure 3-2, the output of the controlling unit is fed into a zero-order hold. The purpose of the zero-order hold is to take the digital bank command signal $\phi(k)$ and convert it to the analog bank command signal $\phi(t)$. A discussion of the operation of a zero-order hold is given in Reference [13]. The conversion of $\phi(k)$ to $\phi(t)$ is simulated by holding the controlling unit output constant for the duration of each sampling period.

Comment on the Random Number Generators

Three random number generators are used in the control system's simulation program. Two are used in the generation of the radar noise and the other in the generation of the wind input. These random number generators possess the ability of repetition, i.e., these random number generators can produce the exact same sequence of random numbers as many times as is needed. With this ability of repetition, the effects of changing portions of the aircraft lateral control system can be studied. This ability will be used in the work presented in Chapter V.

IV. DESIGNING AN OBSERVER

In this chapter an observer will be designed for use in the lateral control system of the F4J aircraft. The observer will be used as a substitute for the tracking α-β filter that is presently being employed. Recall that the tracking α-β filter is used to determine estimates of the lateral position and the lateral velocity of the aircraft from the noisy lateral position radar signal. Therefore the observer will not be designed, as discussed in Chapter II, to reconstruct the aircraft's lateral system state vector, but will be designed to estimate only the aircraft's lateral position and lateral velocity. Block diagrams showing how the α-β filter will be replaced by an observer are given in Figure 4-1 and Figure 4-2.

Equivalent Discrete Reduced Order System

As mentioned above, the function of the observer is to estimate the F4J aircraft's lateral position and lateral velocity. It can be argued that to fulfill this function it is not necessary to construct an observer for the full nine orders of the aircraft lateral system described in Chapter III. Based on this argument a reduced order system will be developed that can be used in the observer design process as a replacement for the F4J aircraft lateral system. The use of this reduced order system will lessen the difficulties associated with the design of an observer for a high order system. Care will be taken when developing this reduced

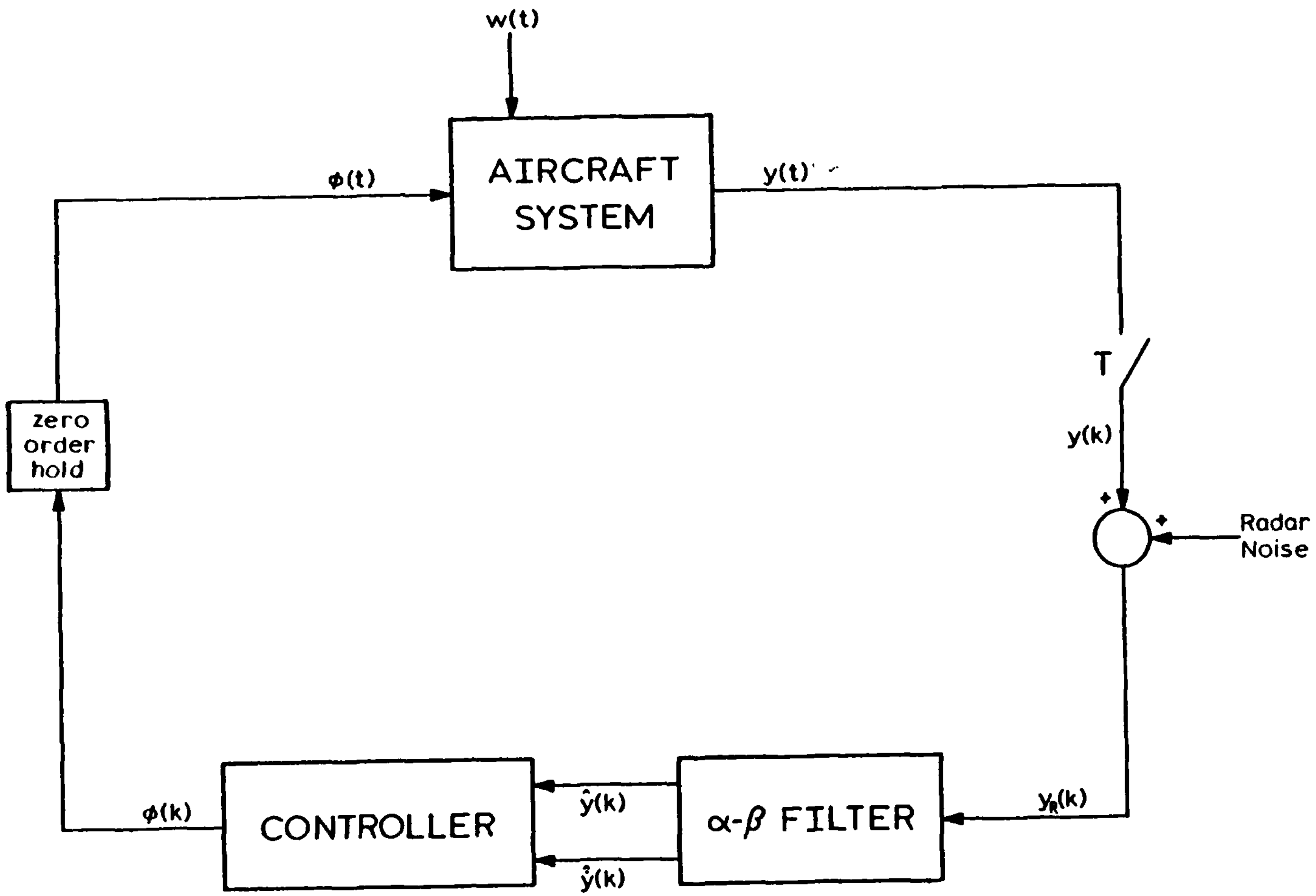


Figure 4-1. Block Diagram of the F4J Aircraft Lateral Control System with the α-β Filter.

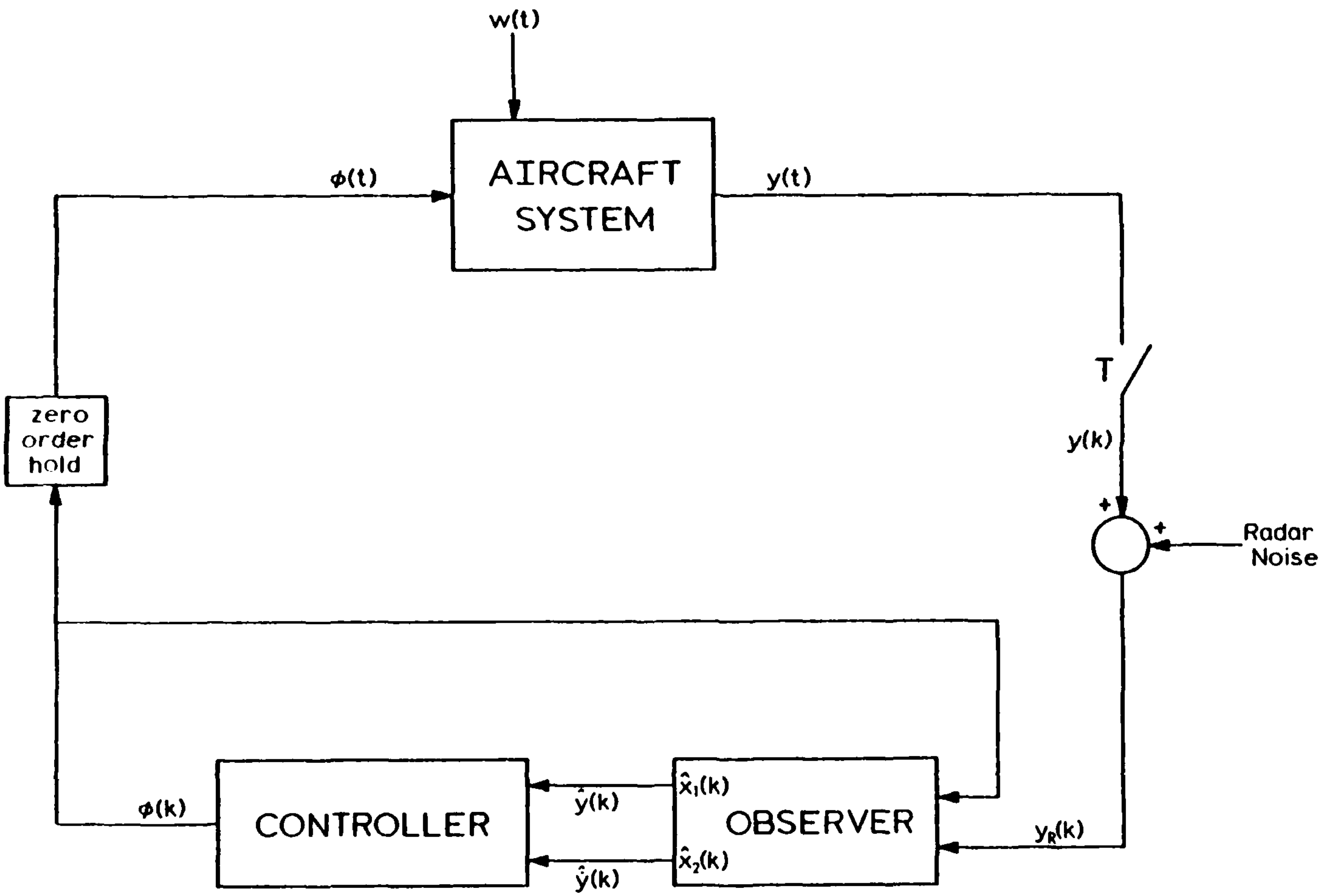


Figure 4-2. Block Diagram of the F4J Aircraft Lateral Control System with the Observer.

order system such that the aircraft's lateral position and lateral velocity are represented in this developed system's state vector.

Once a more manageable system model is developed, it will then be necessary to develop an equivalent discrete model for this reduced order system. This discrete model will be obtained through the use of a procedure that preserves the natural states of the system. Therefore if the aircraft's lateral position and lateral velocity are represented by states in the reduced order system, then they will have the same state representation in the equivalent discrete system. After this discrete system model has been developed, it is then possible to design an observer.

Reduced Order System

The problem of creating a reduced order system to represent a high-order system can be approached in a number of ways. In this paper the desired reduced order system was found by matching frequency responses. The frequency response of a low order system will be matched to the frequency response of the F4J aircraft lateral system.

The aircraft's lateral system frequency response, from bank command input $\phi(t)$ to lateral position output $y(t)$, is computed from a linearized version of the ninth-order state equations given in Chapter III. The most significant portion of this frequency response, that of omega from 0.2 to 2.0, is given in the Bode plot of Figure 4-3. After examining this Bode plot, it was decided that this frequency response could be matched by a third-order system. Through the process of trial and error,

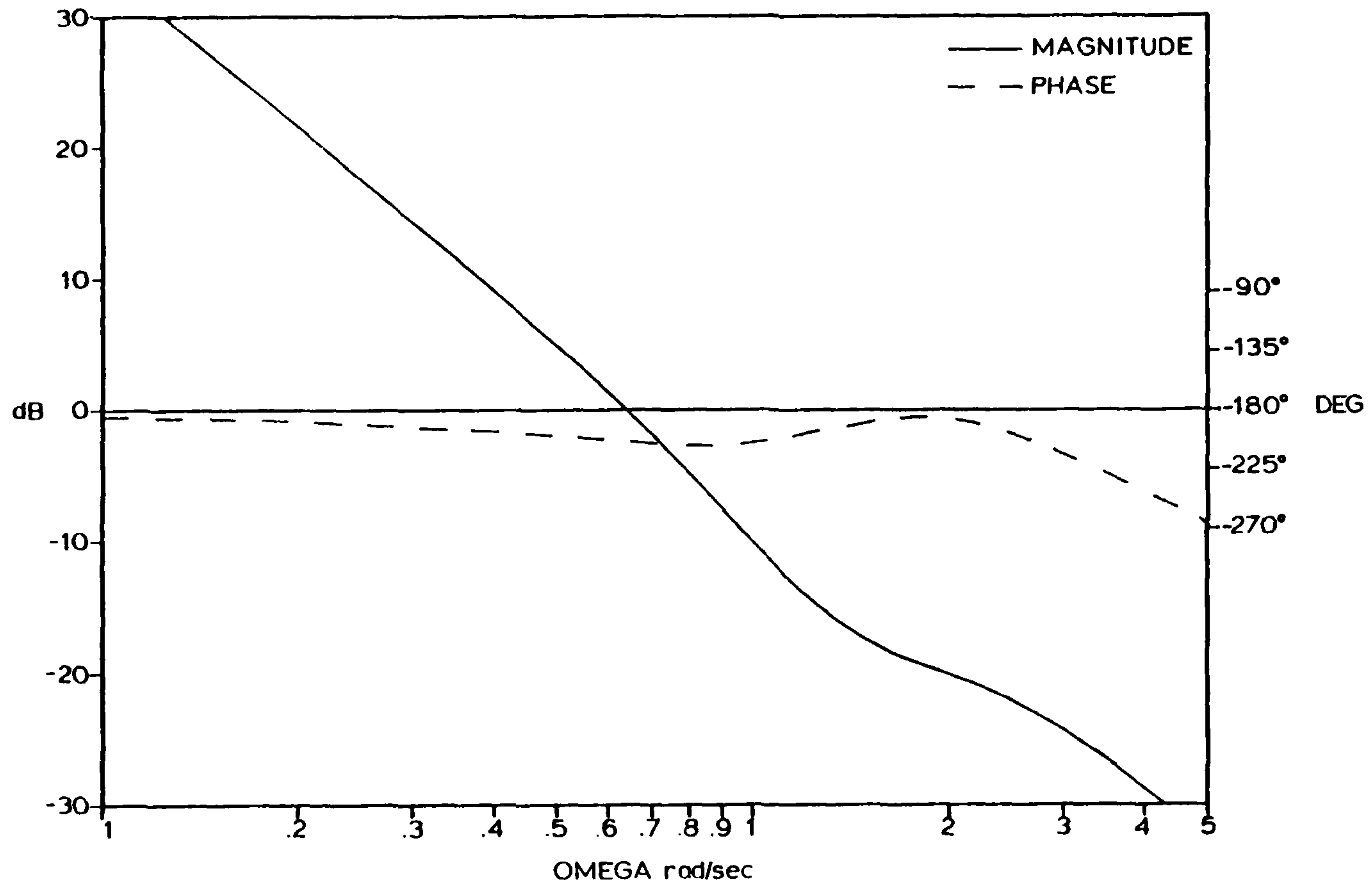


Figure 4-3. Frequency Response of the F4J Aircraft's Lateral System, from Bank Command Input to Lateral Position Output.

a third order system was found that could be used for the desired purpose. A set of continuous state matrix equations of this created system is given in (4-1) and (4-2).

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & -1.42222 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0.0 \\ 0.0 \\ 0.709966 \end{bmatrix} \phi(t) \qquad (4\text{-}1)$$

$$y(t) = \begin{bmatrix} 1.0 & 0.0 & 0.0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} \qquad (4\text{-}2)$$

The frequency response of this third-order system is given in the Bode plot of Figure 4-4. Comparing this frequency response with that of the lateral system given in Figure 4-3, it is concluded that the third order system of (4-1) and (4-2) can be used in the observer design process with small error.

To ensure that the aircraft's lateral position and lateral velocity are represented in the state vector of the reduced order system, a study of this system's state equations will be made. Expanding the equations of (4-1) and (4-2) by matrix multiplication procedures will give the following four differential equations

$$\dot{x}_1(t) = \frac{d}{dt} x_1(t) = x_2(t) \qquad (4\text{-}3)$$

$$\dot{x}_2(t) = \frac{d}{dt} x_2(t) = x_3(t) \qquad (4\text{-}4)$$

$$x_3(t) = \frac{d}{dt} x_3(t) = -1.42222*x_3(t) + 0.709966*\phi(t) \qquad (4\text{-}5)$$

$$y(t) = x_1(t) \qquad (4\text{-}6)$$

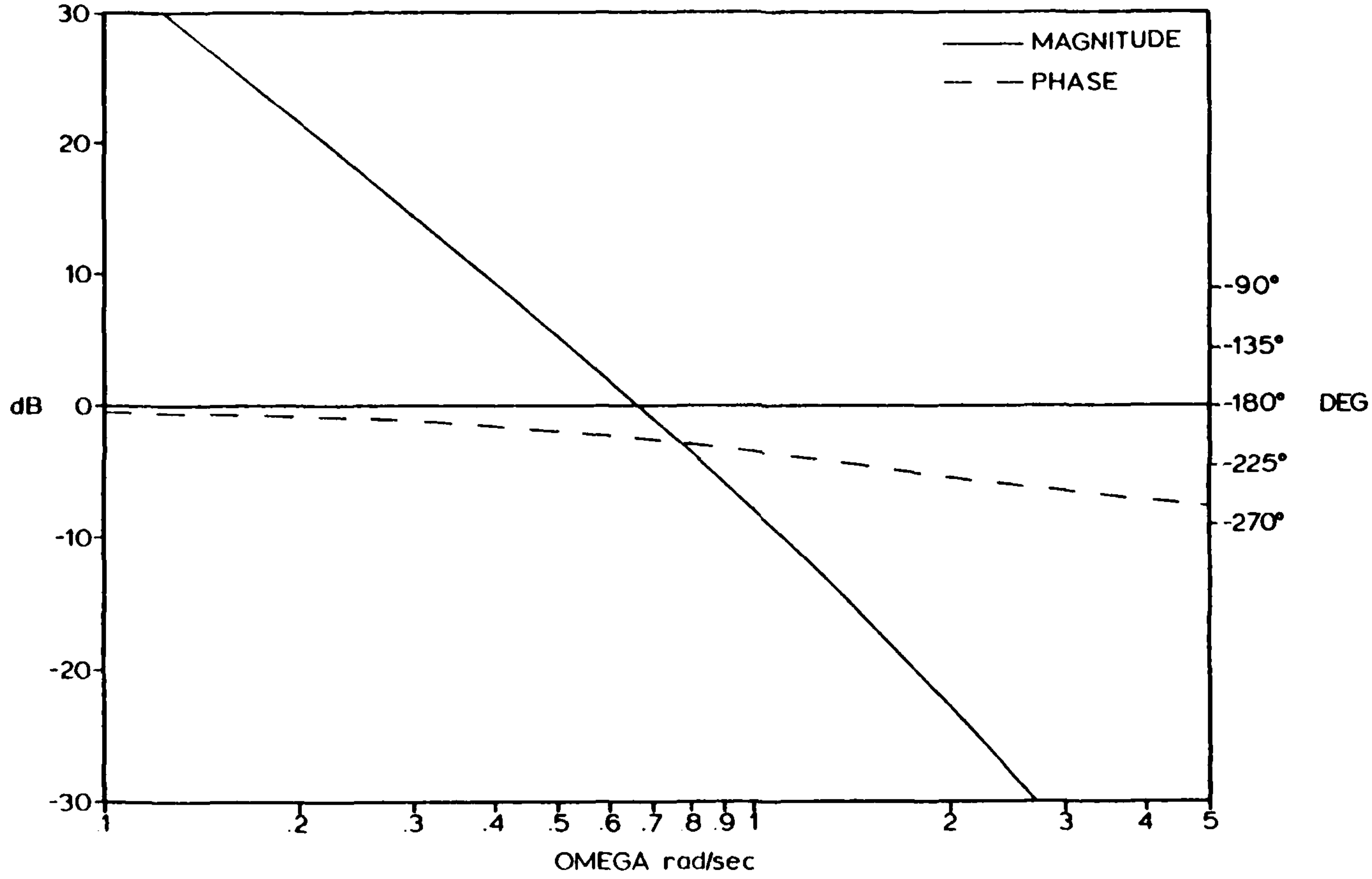


Figure 4-4. Frequency Response of the Reduced Order Representation of the F4J Aircraft's Lateral System, from Bank Command Input to Lateral Position Output.

Since it is known that the output $y(t)$ represents the aircraft's lateral position in the reduced order system, then by (4-6) it is seen that the state $x_1(t)$ must represent the aircraft lateral position. Given this fact, and the fact that the derivative of position with respect to time is velocity, equation (4-3) shows that the state $x_2(t)$ must represent the lateral velocity of the aircraft. Therefore, the aircraft's lateral position and lateral velocity are represented in the state vector of the reduced order system.

To reduce confusion, the continuous state matrix equations, given in (4-1) and (4-2), will be referred to and used as the description of the F4J aircraft lateral system. This will be done until the observer is designed. But it should be noted that the full ninth-order system, with the three nonlinearities and the wind input described in Chapter III, will be used in all simulation runs discussed in the following chapter.

Equivalent Discrete System

In this section, an equivalent discrete model will be developed for the F4J aircraft lateral system. This discrete model will describe the aircraft lateral system combined with the sampler, which is modeled in the radar unit, and the zero-order hold. Therefore the input and the output of this discrete system model will be the digital bank command signal $\phi(k)$ and the sampled lateral position signal $y(k)$, respectively.

The method that is to be used to develop the aircraft lateral system's equivalent discrete model is found in Reference [13]. A BASIC program based on this method is listed in Appendix C. Using this program

an equivalent discrete model for the F4J aircraft's continuous lateral system of (4-1) and (4-2) is developed. This discrete model is given in the following state equations.

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} = \begin{bmatrix} 1.0 & 0.1 & 0.00477116 \\ 0.0 & 1.0 & 0.0932144 \\ 0.0 & 0.0 & 0.867429 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} + \begin{bmatrix} 0.000114237 \\ 0.00338736 \\ 0.0661790 \end{bmatrix} \phi(k) \tag{4-7}$$

$$y(k) = \begin{bmatrix} 1.0 & 0.0 & 0.0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} \tag{4-8}$$

As was mentioned earlier, this equivalent discrete system was developed through the use of a method that preserves the natural states of the system. Therefore, since the states $x_1(t)$ and $x_2(t)$ of the continuous system, (4-1) and (4-2), represent the lateral position and the lateral velocity, respectively, of the aircraft, then the states $x_1(k)$ and $x_2(k)$ of discrete system (4-7) and (4-8) will also represent these same physical variables.

In the next section, an observer will be designed for the discrete third order system given in (4-7) and (4-8). During the design process it will become imperative that the location of this system's poles be known. These pole locations will be determined here.

The poles of a system can be found by evaluating the roots of the characteristic polynomial $\alpha(z)$ of that system. The characteristic polynomial of a system can be found by expanding the following determinate,

$$\alpha(z) = |[zI - A]| \tag{4-9}$$

where the matrix A i the system matrix.

Following this procedure, the pole locations of the F4J aircraft's equivalent discrete lateral system will be found. By substituting the A matrix from (4-7) into (4-9) the characteristic polynomial is obtained.

$$\alpha(z) = \left| \begin{bmatrix} z-1.0 & -0.1 & -0.00477116 \\ 0.0 & z-1.0 & -0.0937144 \\ 0.0 & 0.0 & z-0.867429 \end{bmatrix} \right| \tag{4-10}$$

$$= [(z-1.0)(z-1.0)(z-0.867429)]$$

From this polynomial, the roots, and therefore the pole locations, are apparent. These pole locations are 1.0, 1.0, 0.867429 in the z-plane.

Observer Design

The function of the observer is to extract from the noisy lateral position radar signal tolerable estimations of the F4J aircraft's lateral position and lateral velocity, for use as inputs to the controller. To obtain an observer to fulfill this function, the observer will be designed for the third order system given in (4-7) and (4-8). This third order system is being used to simplify the observer design process; and more importantly, it is being used because the states $x_1(k)$ and $x_2(k)$ of this system are accurate approximations of the lateral position and the lateral velocity if the aircraft. Therefore, the state estimations, $\hat{x}_1(k)$ and $\hat{x}_2(k)$, of the observer designed for this system can be used as the inputs to the controller.

System Observability

Prior to the design of an observer for the aircraft lateral system, it would be prudent to show that this system is observable. This system

will be shown to be observable by establishing that its observability matrix is nonsingular. Note that a matrix is nonsingular if the value of its determinant is nonzero. Recalling the form given in (2-7), the observability matrix θ for the aircraft's lateral system, given in (4-7) and (4-8), is found to be

$$\theta = \begin{bmatrix} 1.0 & 0.0 & 0.0 \\ 1.0 & 0.1 & 0.00477116 \\ 1.0 & 0.2 & 0.0182312 \end{bmatrix} \tag{4-11}$$

The value of this matrix's determinant is computed and is found to be nonzero. Therefore the system of (4-7) and (4-8) is observable.

Design Process

Of the various observer designs discussed in Chapter II, the one that should best function as an estimator of the aircraft lateral position and lateral velocity is the full order observer developed by Franklin and Powell. This observer design was chosen for use over both the reduced-order observer design and the Tse-Athens observer design. The reduced order observer, also developed by Franklin and Powell, cannot be used in the F4J aircraft lateral control system because it was discovered that this lower order observer could not effectively handle the noise that is contained in the radar lateral position signal. The Tse-Athens observer was also eliminated for possible use in the control system because there seemed no feasible method of obtaining the required initial values of the aircraft state vector. Therefore, the observer that is to be designed here, and used in the comparison simulation runs of the next chapter, will be a Franklin-Powell full order observer.

The process of designing a Franklin-Powell full order observer will be started by analyzing the state matrix equation of this observer, given in (2-23),

$$\underline{\hat{x}}(k+1) = [A-LC]\underline{\hat{x}}(k) + B\underline{u}(k) + L\underline{y}(k) \qquad (2\text{-}23)$$

As was discussed in Chapter II, the matrices A, B, and C of this state equation are obtained from the state equations of the system that is to be observed. For this particular process, these matrices will be found in (4-7) and (4-8).

The two vectors, $\underline{u}(k)$ and $\underline{y}(k)$, of equation (2-23) are the inputs to the observer. For the observer being designed here, these two input vectors can be determined by examining the block diagram of Figure 4-2. From this diagram, it is seen that the input vector $\underline{y}(k)$ will be the radar's lateral position signal $y_R(k)$, and the input vector $\underline{u}(k)$ will be the bank command signal $\phi(k)$. After the simulation runs presented in Chapter V were completed, it was discovered that inadvertently $\phi(k-1)$ was used. Fortunately this error was found not to cause any noticeable difference in the results obtained.

The only unknown of the observer state equation, (2-23), is the gain matrix L. Recall from Chapter II that this matrix can be calculated if the locations of the observer poles are known. Therefore, the next step to be taken in this design process is the selection of the pole locations for the observer.

To assist in the selection of proper pole locations for the full order observer, two arguments will now be presented that will provide a description of the desired dynamics of the observer. By knowing the

observer's dynamics, a vague region in the complex z-plane will be outlined. It is within this region that this observer poles should be placed.

The first argument is that the observer dynamics should be "faster" than the dynamics of the F4J aircraft lateral system. The reasoning behind this argument can be found in the discussion of the full order observer error given in Chapter II. As was shown in that discussion, the error vector $\underline{\tilde{x}}(k)$ should converge to zero. This can be accomplished by making the error dynamics, determined from the matrix [A-LC], "faster" than the dynamics of the system that is being observed. It should be noted that the observer dynamics and the error dynamics are equal. Therefore, the observer dynamics should be "faster" than the aircraft's lateral system dynamics so that the error will tend to vanish.

The second argument is based on the noise that is contained in the radar lateral position signal. There is such a significant amount of noise in this signal that caution should be taken in the determination of how "fast" the observer dynamics should be made. Large errors will be created in the state estimations if the observer dynamics are so "fast" that the observer reacts more to the noise than to the lateral movements of the aircraft. Therefore, according to this argument, the observer's dynamics should be "slow", so that there is no overreaction to the radar noise.

Combining these two arguments, a description of the observer's desired dynamics can be derived. The dynamics of the observer should be only slightly "faster" than the F4J aircraft's lateral system dynamics. From this description and by noting the relationship between a system's dynamics and the location of its poles in the complex z-plane, the

observer pole locations can be selected. This relationship states that the closer to the z-plane origin that a system's poles are located, the "faster" the system's dynamics will become. Therefore, the full order observer poles must be placed closer to the z-plane origin than those poles of the aircraft lateral system, but not so close that the noise will dominate.

As was found in the previous section, the three poles of the F4J aircraft lateral system, (4-7) and (4-8), are located at 1.0, 1.0, 0.867429 in the z-plane. Hence the pole locations of the full order observer are selected to be at 0.8, 0.8, 0.8 in the z-plane. As can be seen, all three poles are chosen to be closer to the z-plane origin than any pole of the aircraft lateral system.

Now that the location of the observer's poles have been selected, the unknown gain matrix L can be computed. Using the BASIC computer program of Ackermann's Formula, given in Appendix B, the L matrix for this full order observer is obtained.

$$L = \begin{bmatrix} 0.467428 \\ 0.57863 \\ 0.0353145 \end{bmatrix} \tag{4-12}$$

Substituting this gain matrix L, and the matrices A, B, and C for (4-7) and (4-8), into their respective positions of the state equation (2-23) results in state matrix equation for the full order observer.

$$\begin{bmatrix} \hat{x}_1(k+1) \\ \hat{x}_2(k+1) \\ \hat{x}_3(k+1) \end{bmatrix} = \begin{bmatrix} 0.532572 & 0.1 & 0.00477116 \\ -0.57863 & 1.0 & 0.0432144 \\ -0.0353145 & 0.0 & 0.867424 \end{bmatrix} \begin{bmatrix} \hat{x}_1(k) \\ \hat{x}_2(k) \\ \hat{x}_3(k) \end{bmatrix} + \begin{bmatrix} 0.00014237 \\ 0.00338736 \\ 0.0661740 \end{bmatrix} \phi(k) + \begin{bmatrix} 0.467428 \\ 0.57863 \\ 0.0353145 \end{bmatrix} y_R(k) \qquad (4\text{-}13)$$

This observer will now be used as a substitute for the tracking α-β filter in the F4J aircraft's lateral control system. To study the effects of this substitution, a supplement to the control system's simulation program, listed in Appendix D, was written for the observer's state equation of (4-13).

V. RESULTS

The possibility of using the observer of (4-13) as a substitute for the tracking α-β filter in the F4J aircraft lateral control system is investigated in this chapter. This investigation will be accomplished in two steps. First the acceptability of the observer as a substitute for the α-β filter will be shown. Second the performance of the observer when used in the lateral control system will be judged. These steps will be executed by comparing responses of the control system with the α-β filter used to estimate the aircraft lateral position and lateral velocity, to similar responses of the control system with the observer used to produce the estimations. Block diagrams of these control systems were given in Figure 4-1 and Figure 4-2. For the remainder of this discussion these two lateral control systems will be referred to as the α-β filter control system and the observer control system. The responses of these systems, that are to be presented here, were obtained through the use of the system simulation as described in Chapter III. Also in this chapter, a technique for selecting other pole locations for an observer used in the lateral control system is developed.

Observer Acceptability

The acceptability of the observer given in (4-13) as a substitute for the tracking α-β filter will be shown through a comparison of two sets of responses of the α-β filter control system and the observer

control system. These responses will be the open-loop frequency response, and the time response of the system with a given initial condition. From these responses, basic system characteristics can be determined that will be used in the comparison. These characteristics for both control systems are given in Table 5-1. The responses and the characteristics of the α-β filter control system are assumed to be satisfactory for the lateral control system in the context of this paper. Therefore, if the responses and the characteristics of the observer control system compare favorably to those of the α-β filter control system, then the observer of (4-13) will be considered to be an acceptable substitute for the tracking α-β filter. A brief discussion of the responses and how the system characteristics were determined from the responses is given below.

Open-Loop Frequency Response

The open-loop frequency responses of the α-β filter control system and the observer control system are given in the Bode plots of Figure 5-1 and Figure 5-2. These frequency responses were computed from linearized models of the lateral control system. A comparison of these plots shows that the open-loop frequency responses of the two systems are fairly similar.

The stability margins of the two systems are determined from the open-loop frequency responses shown in Figure 5-1 and Figure 5-2. The gain margin, i.e., the amount that the gain must be increased to cause the system to become unstable, is determined by noting the magnitude of the system's gain when the phase angle is -180°. The phase margin, i.e., the amount of phase lag that must be added to the system to cause instability, is equal to the phase angle of the system at unity gain, plus

Table 5-1.

System Characteristics.

	α-β Filter Control System	Observer Control System
Gain Margin, GM (dB)	15	12
Phase Margin, PM (degrees)	49	46
Time to Rise, Tr (seconds)	8	8
Time to Peak, Tp (seconds)	17	17
Time to Settle, Ts (seconds)	52	53
Percent Overshoot, P.O. (%)	40	44

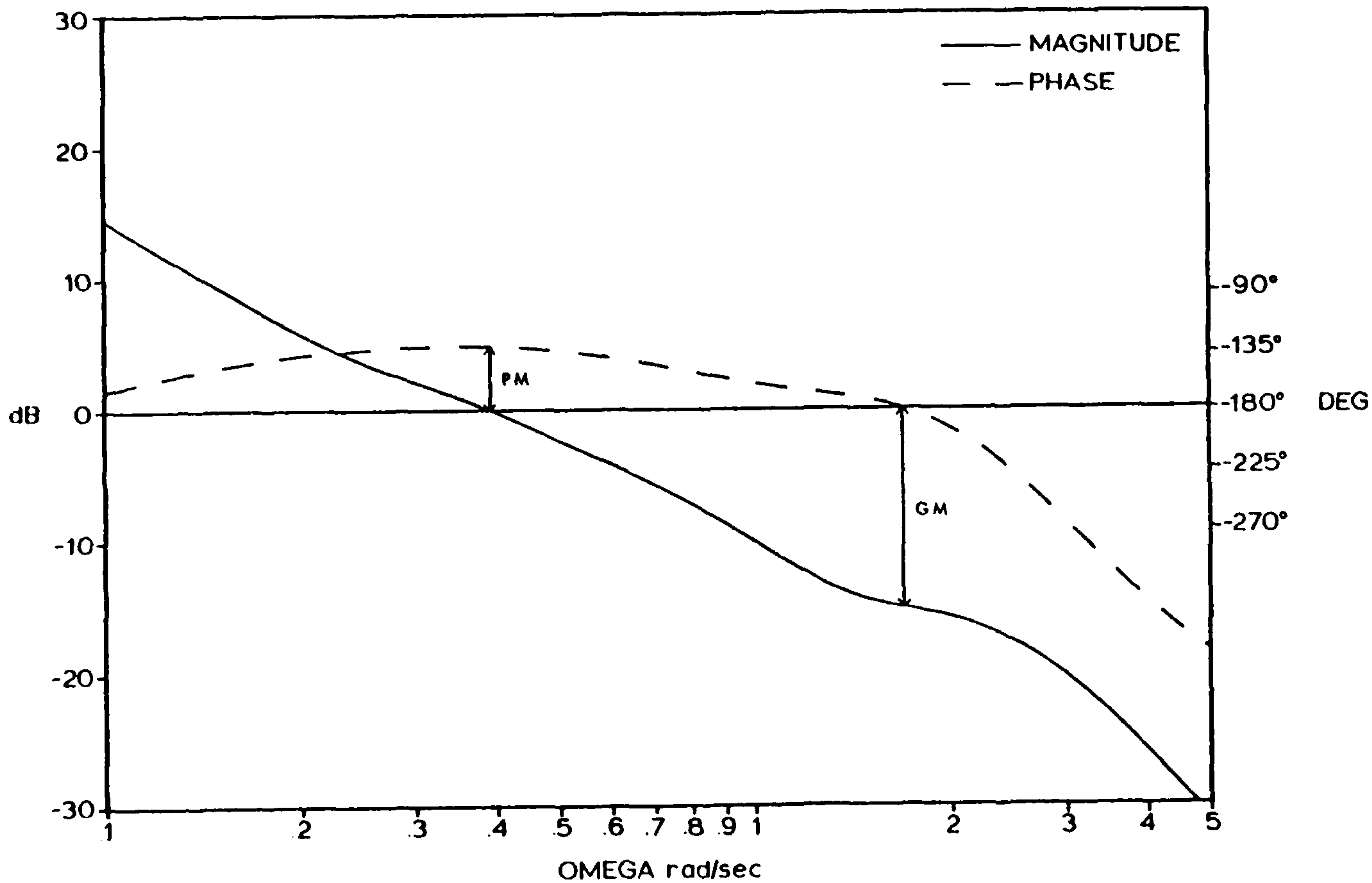


Figure 5-1. Open-Loop Frequency Response of the α-β Filter Control System.

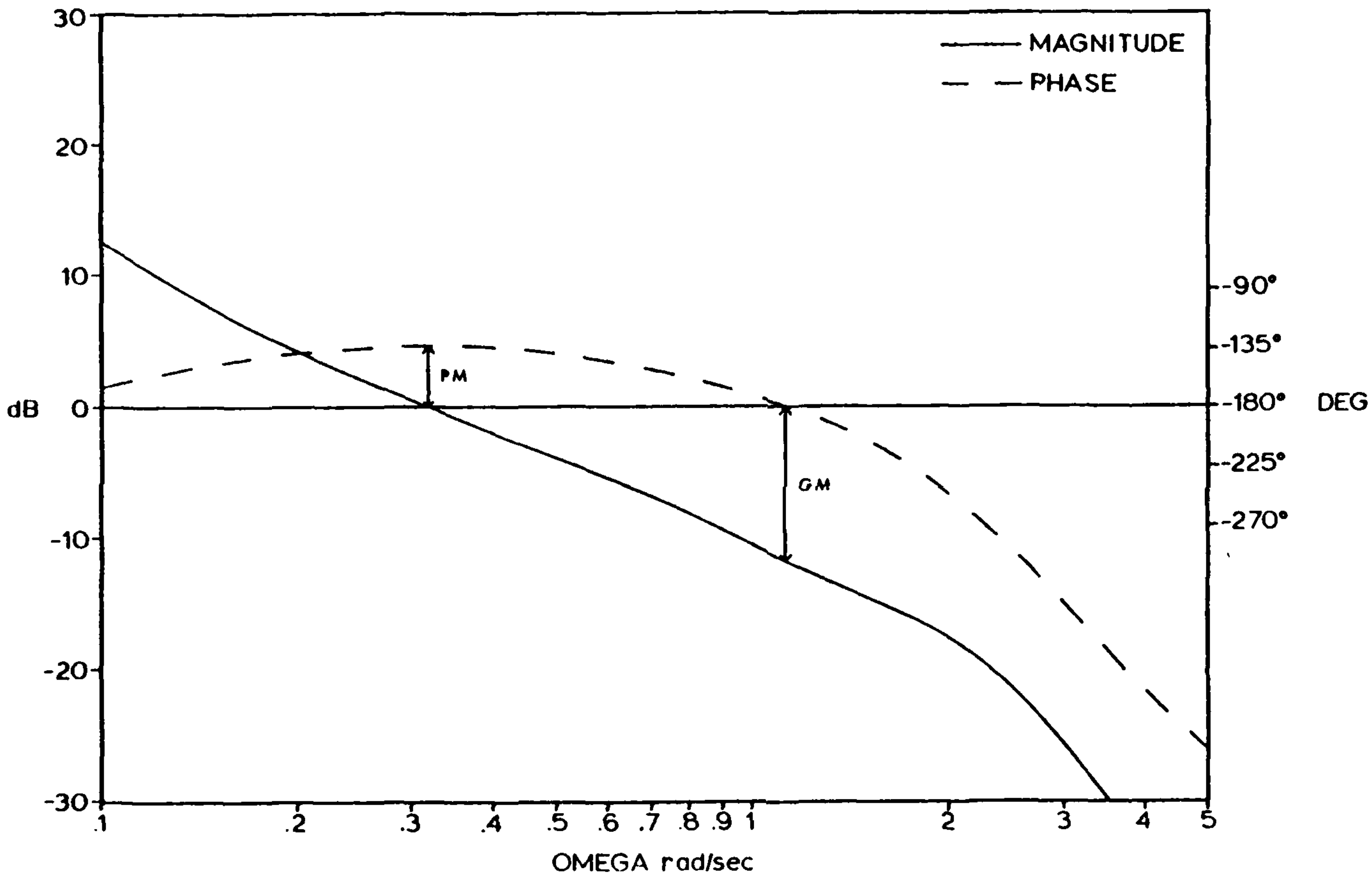


Figure 5-2. Open-Loop Frequency Response of the Observer Control System.

180°. As can be seen from Table 5-1, these characteristics of the two lateral control systems compare favorably.

Time Response with a Given Initial Condition

The initial-condition time responses of the α-β filter control system and the observer control system are given in Figure 5-3 and Figure 5-4. These time responses are from simulation runs of the last sixty seconds of the aircraft's flight before touchdown. Prior to these final sixty seconds, the aircraft is assumed to be twenty feet laterally off the extended centerline of the runway, with no lateral movements. The forward velocity of the aircraft is considered to be a constant of 220.39 feet per second. The wind and the radar noise disturbances have been eliminated in these simulation runs. As can be seen by comparing Figure 5-3 and Figure 5-4, the time responses of the two lateral control systems, with a given initial condition, are almost identical.

The system characteristics determined from the initial-condition time responses are the final four characteristics given in Table 5-1. The time to rise and the time to peak characteristics give a measure of the speed of the system response. These characteristics are defined as the time the aircraft requires to reach the runway's extended centerline, and the time needed to reach its first peak, respectively. The time to settle and the percent overshoot characteristics give a measurement as to how well the lateral control systems guide the aircraft. The time to settle is determined by determining the amount of time necessary for the system to settle the aircraft within 2% of its initial condition of 20 feet. The percent overshoot is calculated by

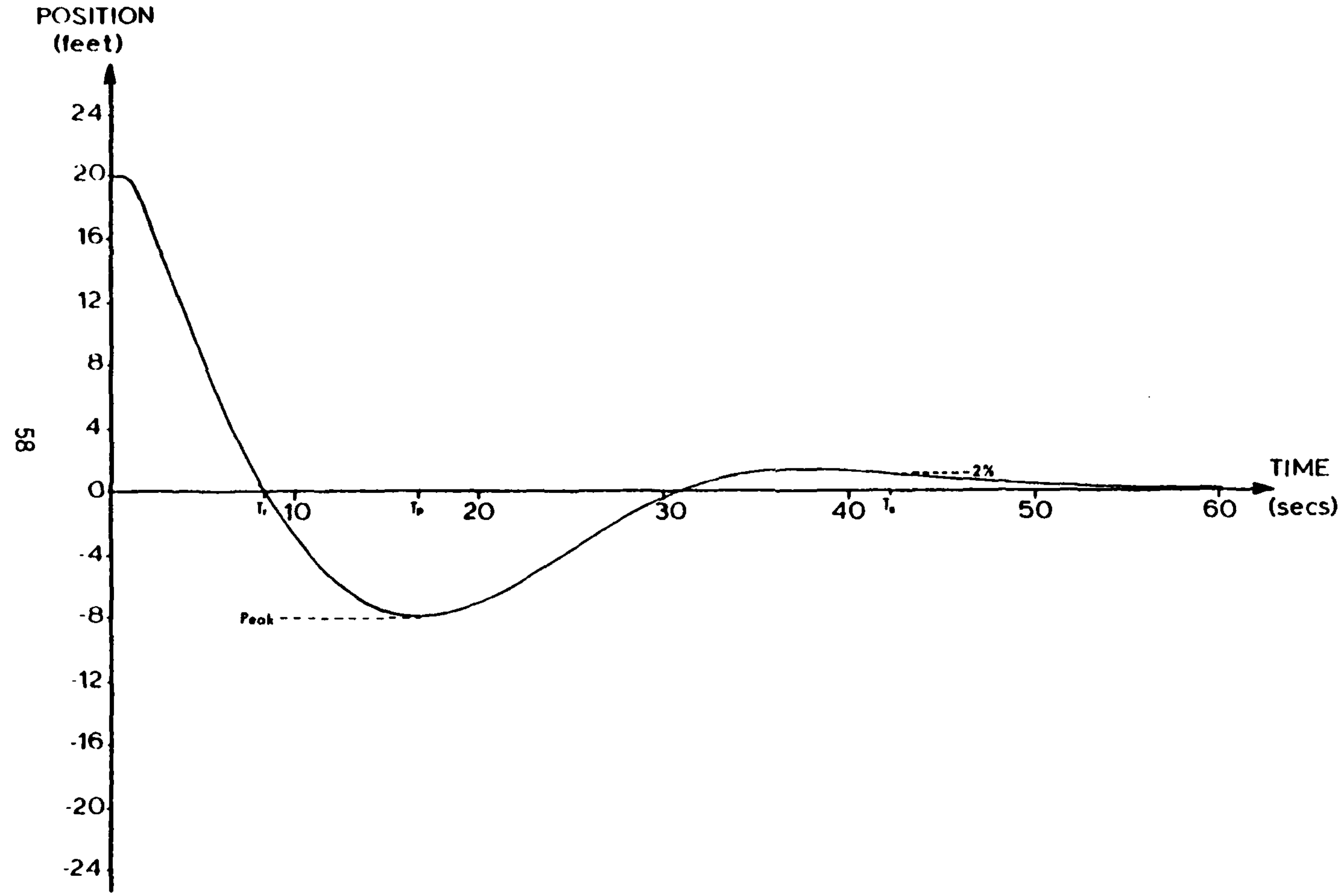


Figure 5-3. Initial Condition Time Response of the α-β Filter Control System.

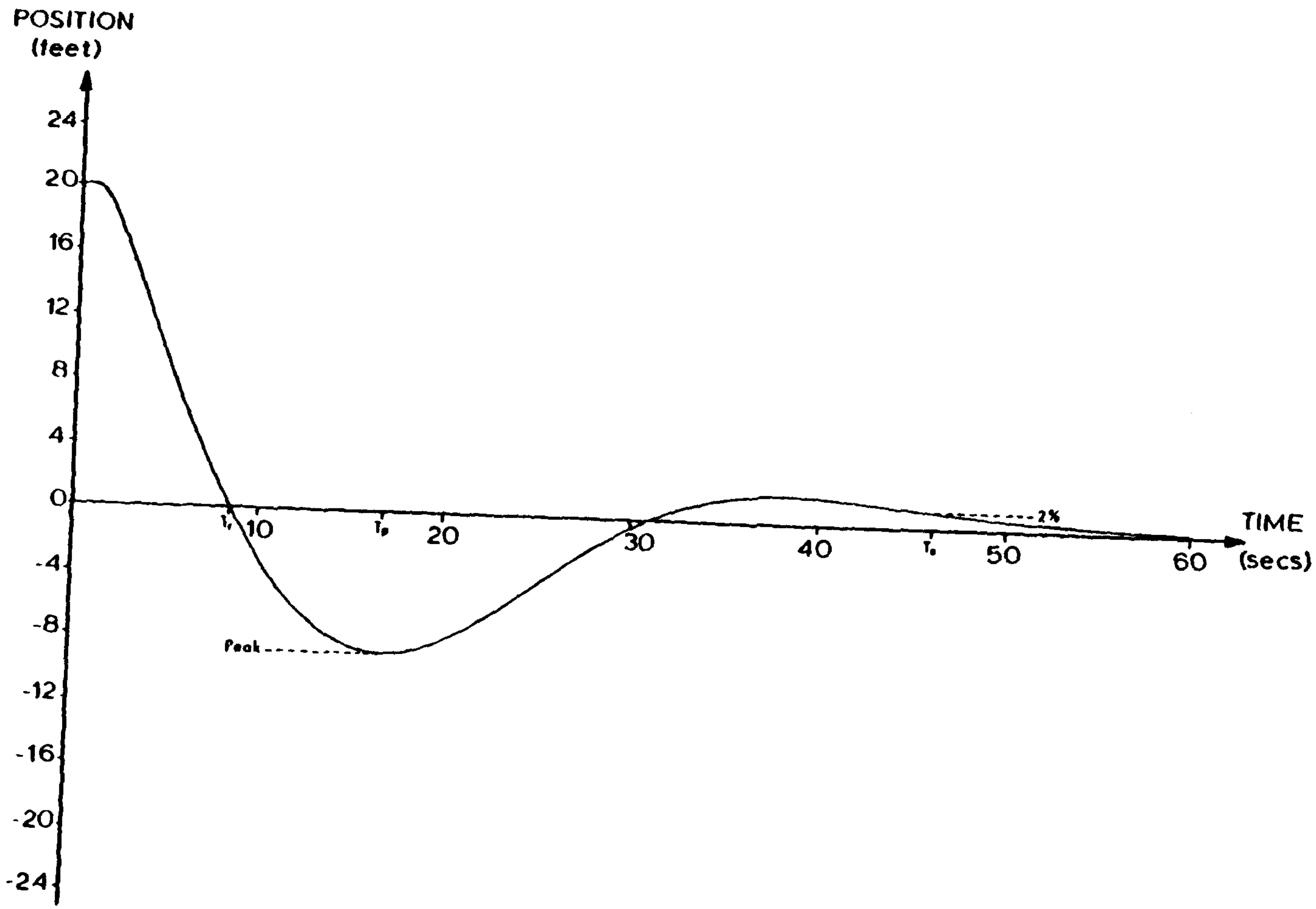


Figure 5-4. Initial Condition Time Response of the Observer Control System.

$$P.O. = \frac{Peak}{I.C.} \times 100\% \qquad (5\text{-}1)$$

where Peak is the magnitude of the first peak of the time response, and I.C. is the magnitude of the initial condition. Note that the four sets of system characteristics given are quite similar.

Therefore, by the comparison of the two sets of responses, along with the associated system characteristics, it is assumed here that the observer given in (4-13) is an acceptable substitute for the tracking α-β filter in the F4J aircraft lateral control system.

Observer Performance

The performance of the observer of (4-13) when used in the lateral control system of the F4J aircraft will now be judged. This judgement will be based on the results of a number of Monte Carlo runs of the observer control system as compared to results of similar runs of the α-β filter control system. After these comparisons have been made, an investigation will be presented to show why the observer control system performed differently from the α-β filter control system.

Monte Carlo Runs

Monte Carlo runs are used in this paper to give statistical measurements to the performance of the two lateral control systems. Each of the Monte Carlo runs that are to be presented were determined from twenty simulation runs. Each of these simulation runs are of the last sixty seconds of the aircraft's flight before touchdown. Prior to the start of each simulation run the aircraft is assumed to be in lateral steady-state flight, with a forward velocity of 220.39 feet per second, along the

centerline of the runway. All lateral movements of the aircraft during the simulation runs are caused by the wind and/or the radar noise. For each of the twenty simulation runs, the three random number generators of the wind input and the radar noise input were forced to generate completely different sequences of random numbers. But, it should be noted that each Monte Carlo run used identical sets of random number sequences.

The results of the Monte Carlo runs of the two control systems, with various combinations of wind and radar noise, are listed in Table 5-2. The results presented in this table are the average values and the r.m.s. values of the aircraft's lateral position error off the extended centerline of the runway, over the twenty simulation runs. More importance is given to the r.m.s. values than the average values, since in calculating the average values a large position error to one side of the centerline could be compensated for by a large position error to the other side of the centerline. A list of the percent improvement of the observer control system over the α-β filter control system is also given. This percent improvement was calculated by

$$\% \text{ Improvement} = \frac{(R_F - R_O)}{[(R_F + R_O)/2]} \times 100\% \qquad (5\text{-}2)$$

where R_F and R_O are the r.m.s. values of the α-β filter control system and the observer control system respectively.

The results in Table 5-2 indicates that the observer control system reduces the effects of the radar noise disturbance on the controlling of the lateral movements of the aircraft better than does the α-β filter control system. The reverse is true, on a smaller scale, with respect to the wind distrubances. With both of the disturbances included, it is

Table 5-2.

Results of Monte Carlo Runs.

	α-β Filter Control System		Observer Control System		% Improvement
	average	r.m.s.	average	r.m.s.	
Wind and Radar Noise	-0.0030954	5.6082811	-0.2171728	3.9342303	35.1%
Wind Only	-0.0950927	1.9006910	-0.0875373	2.1553383	-12.6%
Radar Noise Only	-0.2798429	2.5217228	-0.0506595	1.6845322	39.6%

seen that the observer control system gives better results than the α-β filter control system. Therefore, from these results the observer of (4-13) is judged to give a better performance when used in the lateral control system of the F4J aircraft than the tracking α-β filter that is presently being employed. An investigation of why and how the observer control system gives better results will now be discussed.

Closed-Loop Frequency Response

A possible reason for the observer control system to perform better than the α-β filter control system, with respect to the radar noise, can be determined from an examination of the closed-loop frequency responses of the two lateral control systems. The closed-loop frequency responses of the α-β filter control system and the observer control system, from the radar noise input to the lateral position output, are given in the Bode plots of Figure 5-5 and Figure 5-6. These frequency responses were computed from linearized versions of the lateral control systems.

The -3dB bandwidths of the lateral control systems are determined from the closed-loop frequency responses. The α-β filter control system bandwidth is found to be 0.77 radians/second while the bandwidth of the observer control system is determined to be 0.68 radians/second. Since the observer control system has a narrower bandwidth, the time response of this control system will be slower than the time response of the α-β filter control system. In other words, the observer control system is less sensitive to the radar noise disturbance than is the α-β filter control system.

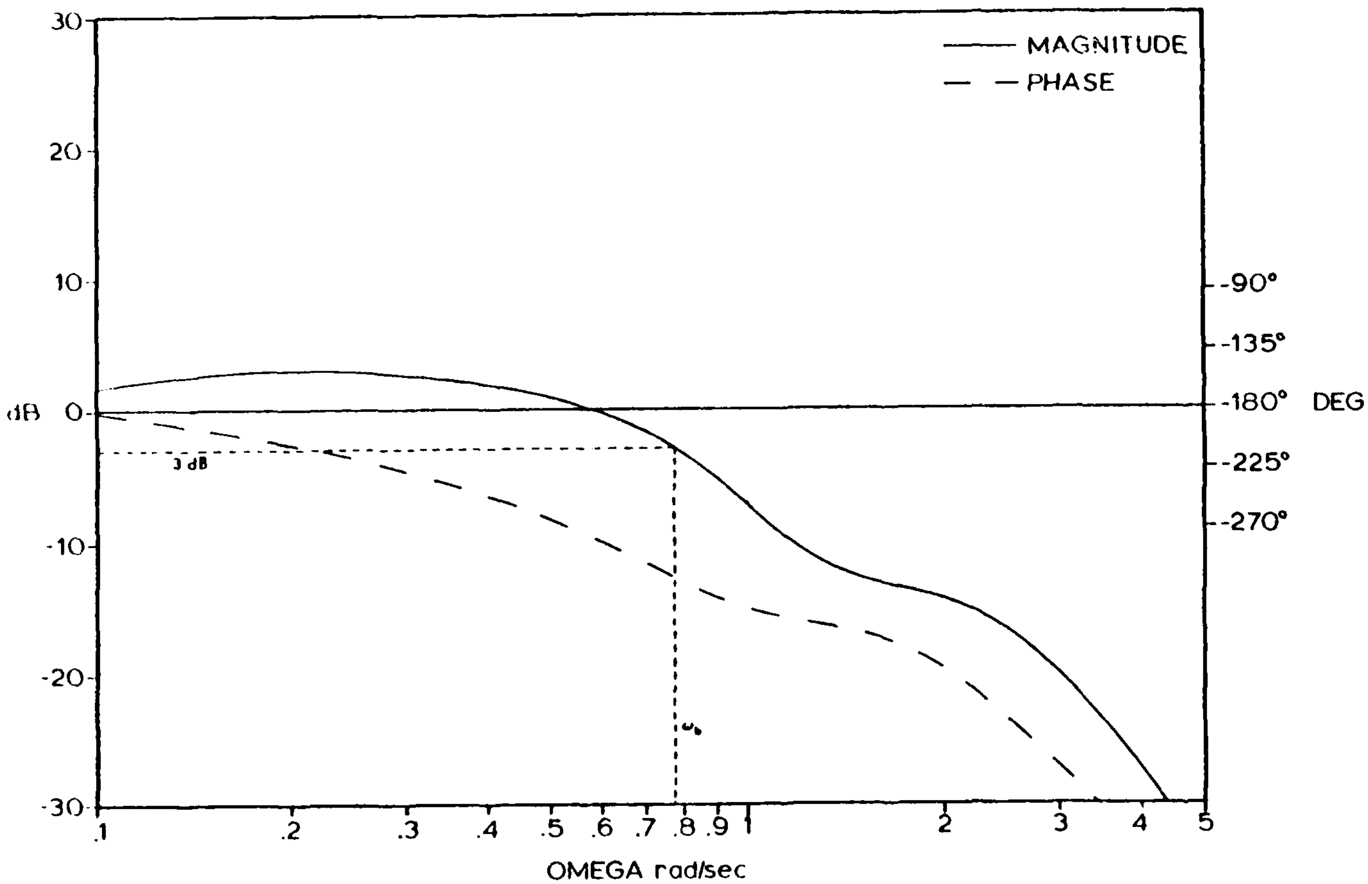


Figure 5-5. Closed-Loop Frequency Response of the α-β Filter Control System, from Radar Noise Input to Lateral Position Output.

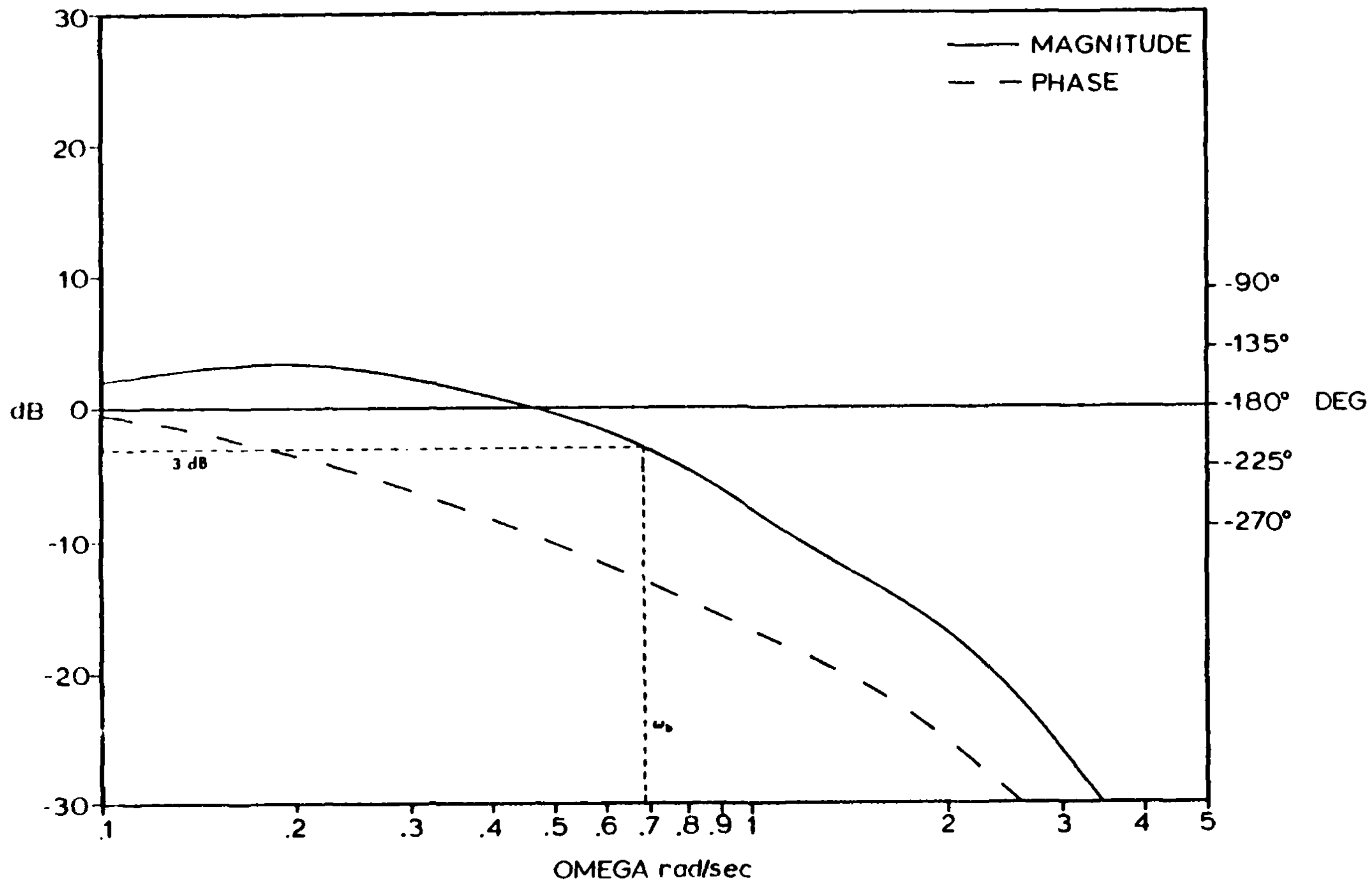


Figure 5-6. Closed-Loop Frequency Response of the Observer Control System, from Radar Noise Input to Lateral Position Output.

Simulation Runs

It has been shown that the observer control system achieved a better performance than the α-β filter control system when both the effects of the wind and the radar noise are included. The ensuing discussion will attempt to explain the observer control system's better performance through detailed examinations of comparable simulation runs of the two lateral control systems. These simulation runs are taken from the twenty simulation runs used in the Monte Carlo analysis. It should be noted that identical wind and radar noise disturbances were used in both simulation runs.

The results of the examination of the two lateral control system simulation runs are illustrated in the time responses given in Figure 5-7 through Figure 5-11. The responses of the two simulation runs are given in Figure 5-7. The improvement in the control of the aircraft when the observer is used to estimate the aircraft's lateral position and lateral velocity is shown in a comparison of the two responses.

Shown in Figure 5-8 is a comparison of the actual lateral position of the aircraft to the estimates of this position, that were produced by the observer and the α-β filter. These responses were obtained from the simulation run of the observer control system. Figure 5-9 shows a similar comparison, except that these responses were obtained from the simulation run of the α-β control system. From these two figures, it can be seen that the observer and the α-β filter estimate the aircraft's lateral position to an approximately equal quality. Hence the improvement in

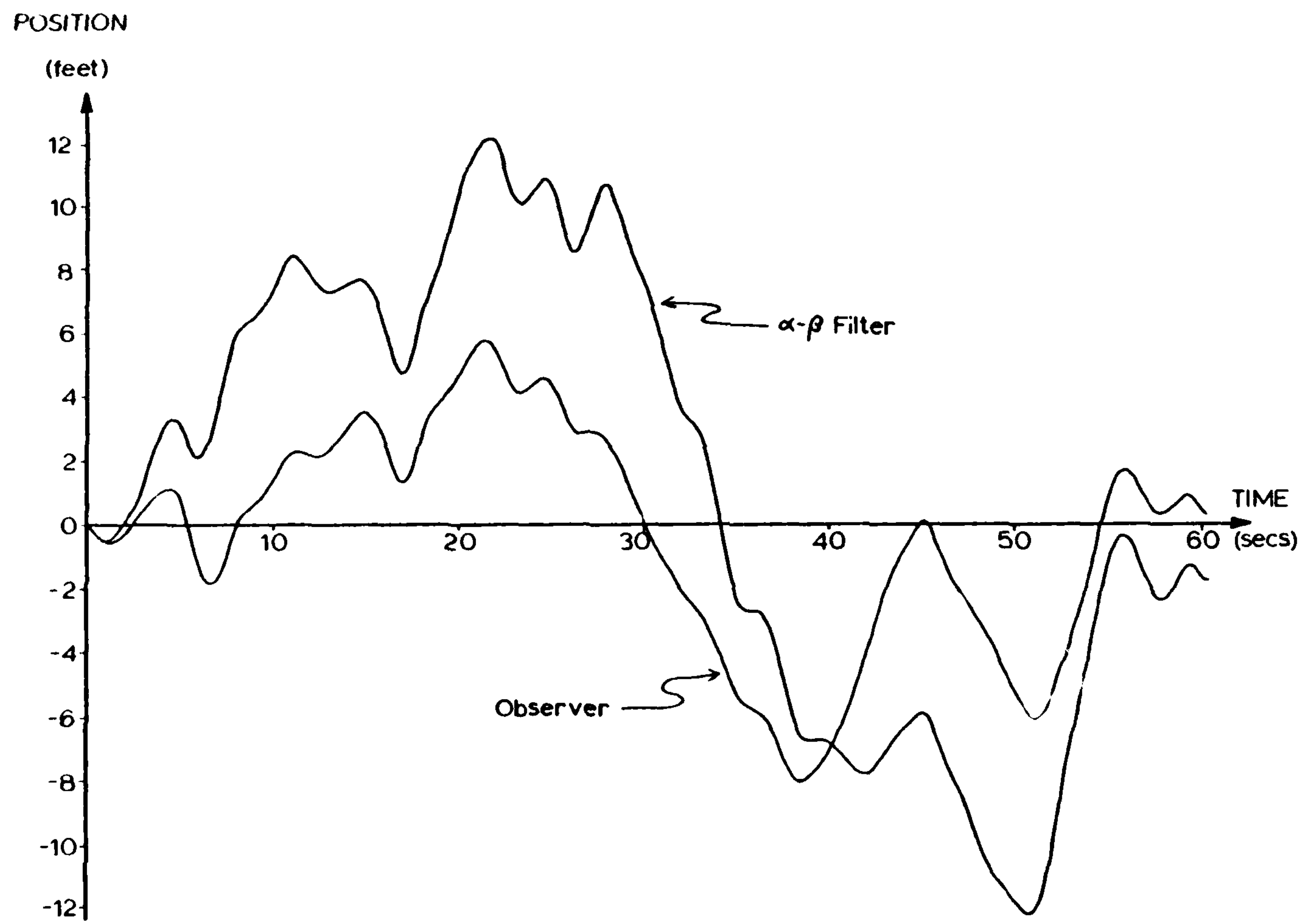


Figure 5-7. Response of the Observer Control System Compared to the Response of the α-β Filter Control System, with the Same Wind and Radar Noise Disturbances.

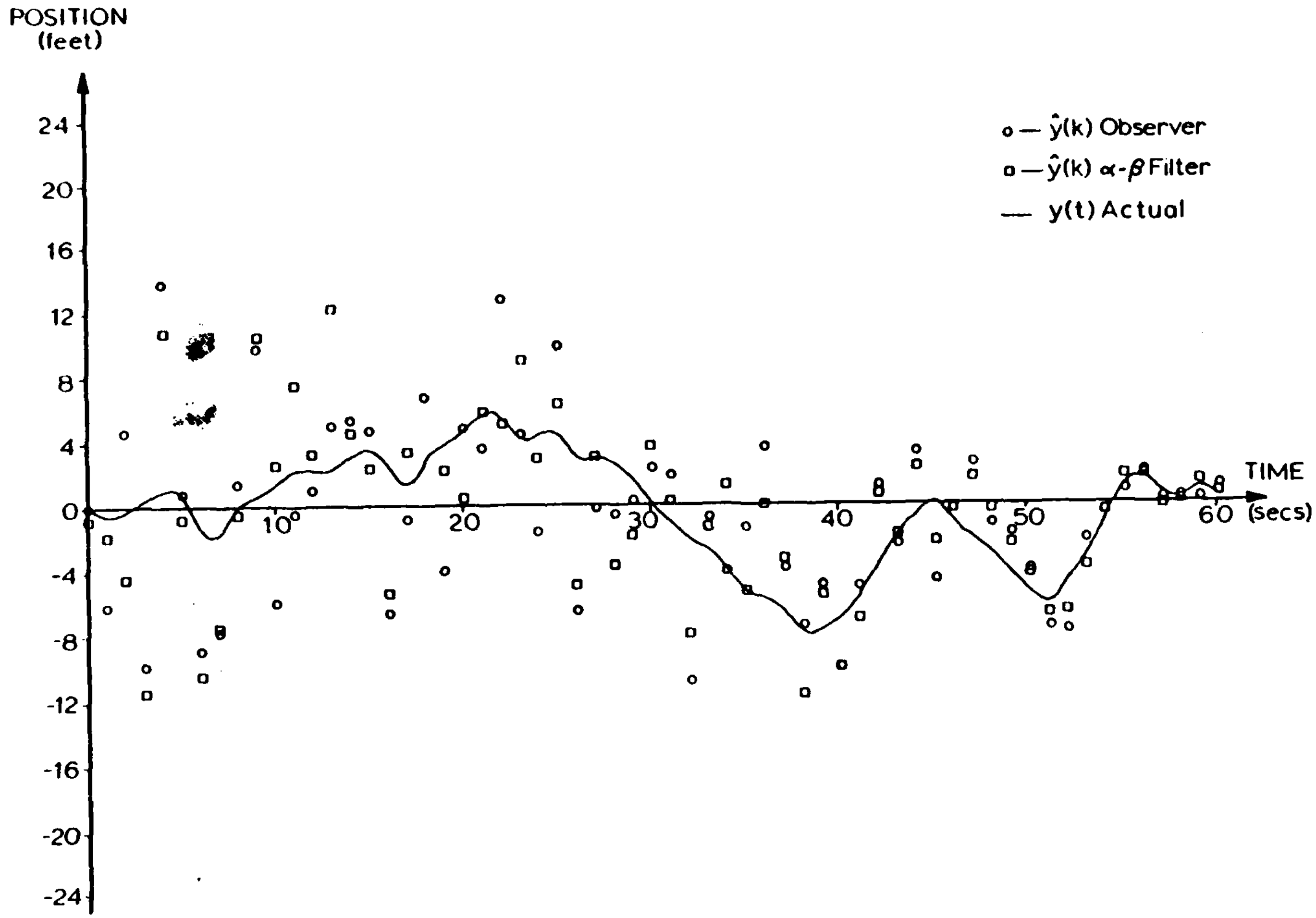


Figure 5-8. Lateral Position of the Aircraft Controlled by the Observer Control System Compared to the Estimates Produced by the Observer and the α-β Filter.

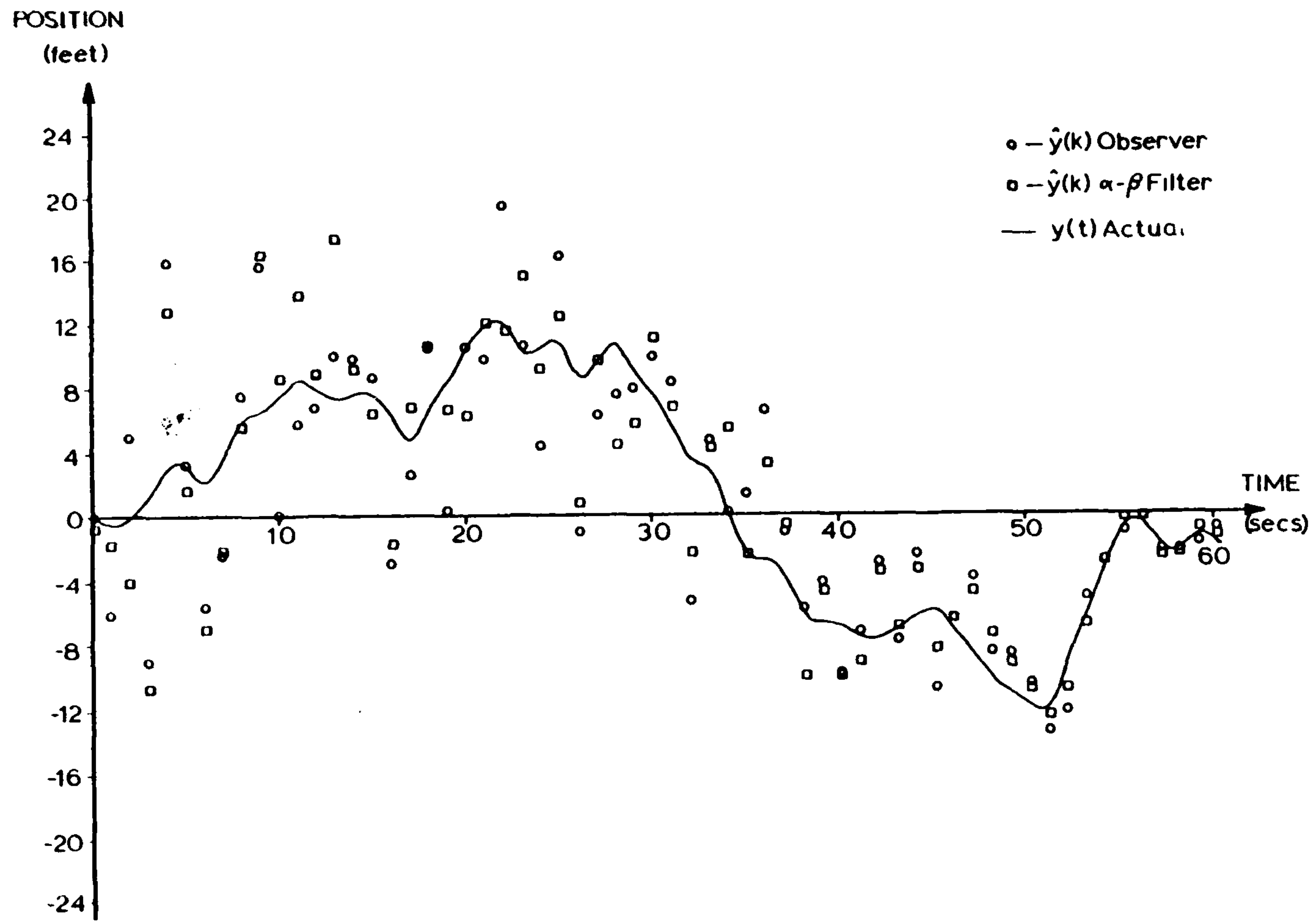


Figure 5-9. Lateral Position of the Aircraft Controlled by the α-β Filter Control System Compared to the Estimates Produced by the Observer and the α-β Filter.

the performance that the observer control system has over the α-β filter control system is not obtained from the estimation of the aircraft's lateral position.

A similar examination of the aircraft's lateral velocity, and it's estimates, is given in Figure 5-10 and Figure 5-11. As is shown in these figures, the observer estimates the lateral velocity more accurately than the α-β filter. Therefore, the improvement in the performance that the observer control system has over the α-β filter control system is concluded to be obtained from the estimation of the aircraft's lateral velocity.

Selecting Other Observer Pole Locations

A technique to select other pole locations for an observer used in the lateral control system of the F4J aircraft is presented in this section. This technique will select the location of the observer's poles by determining a gain matrix that will reduce the effects of the radar noise on the estimates of the aircraft's lateral position and lateral velocity. But before this technique can be presented, the observer developed in Chapter IV must be corrected.

Recall that the error in the simulation of the observer, given in (4-13), is that the bank command input is delayed by one sampling period with respect to the remainder of the observer. To correct this error, a delayed version of the observer equation must be used. The form of this delayed observer state matrix equation will be

$$\underline{x}(k) = [A-LC]\underline{x}(k-1) + B\phi(k-1) + Ly_R(k-1) \qquad (5\text{-}3)$$

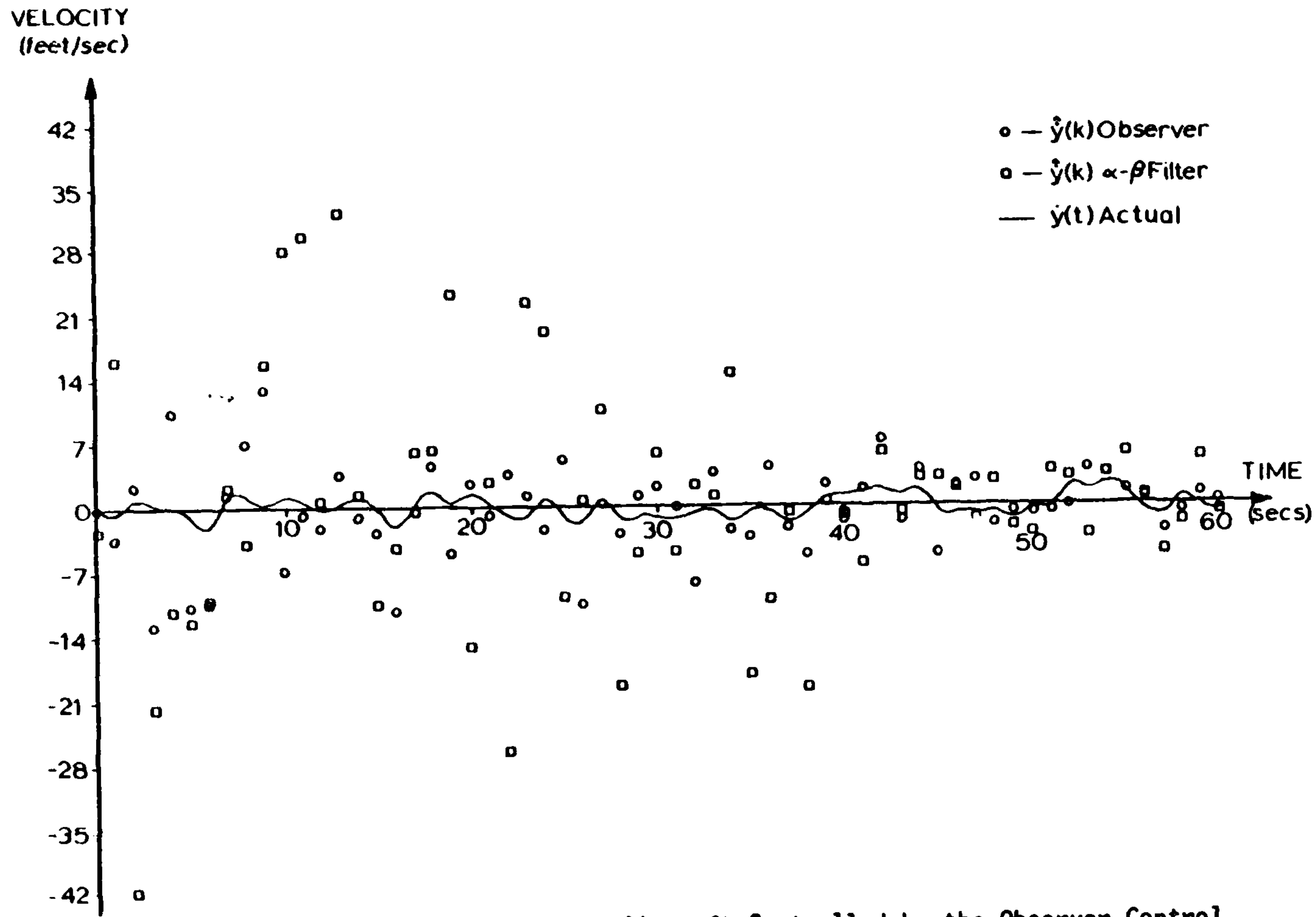


Figure 5-10. Lateral Velocity of the Aircraft Controlled by the Observer Control System Compared to the Estimates Produced by the Observer and the α-β Filter.

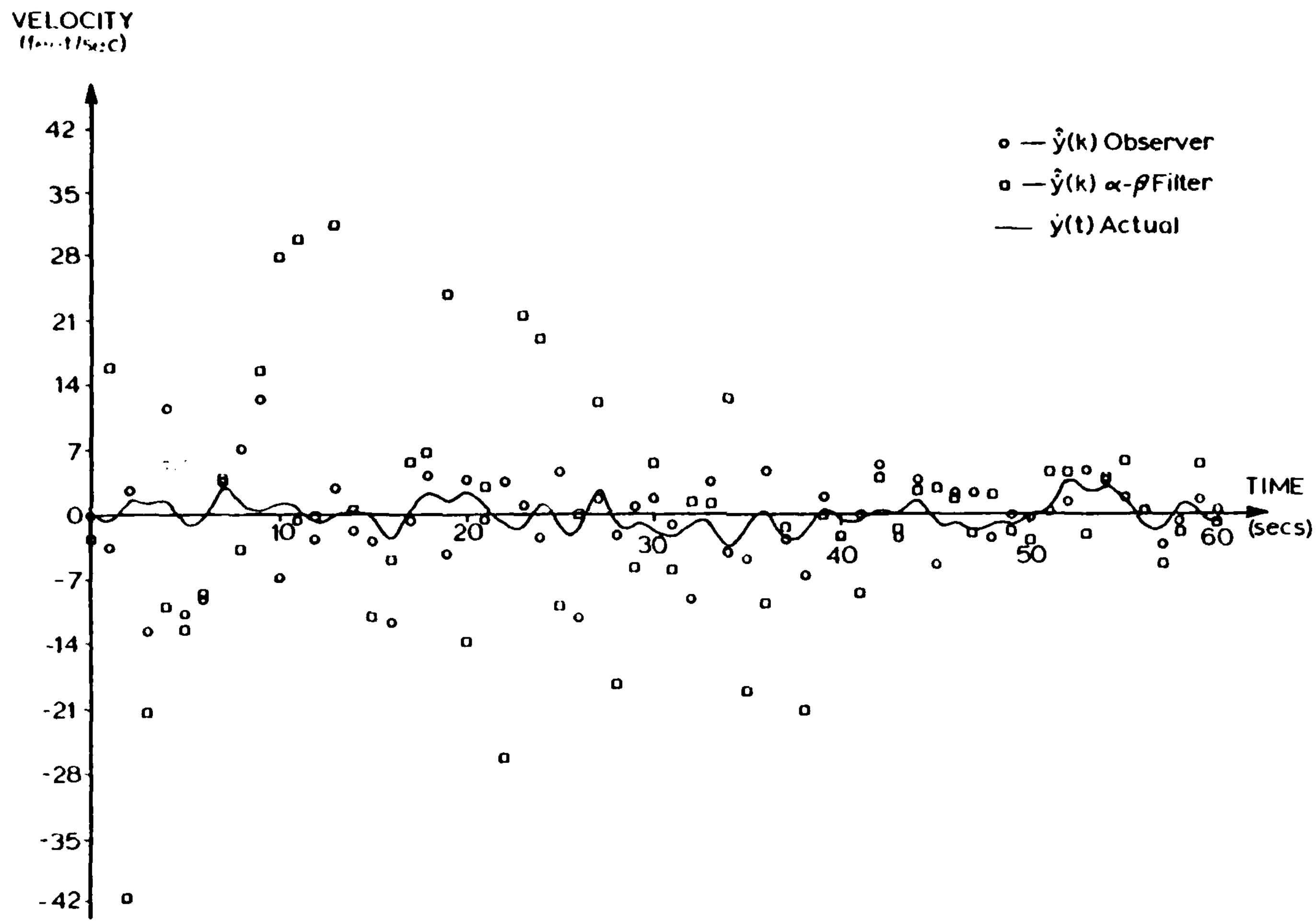


Figure 5-11. Lateral Velocity of the Aircraft Controlled by the α-β Filter Control System Compared to the Estimates Produced by the Observer and the α-β Filter.

The matrices A, B, C, and L are as they were described in Chapter IV and the inputs $\phi(k-1)$ and $y_R(k-1)$ are the delayed bank command and the lateral position radar signals. This corrected observer is illustrated in the block diagram of Figure 5-12. With the observer corrected, the development of the technique of selecting other pole locations can be continued.

To determine a gain matrix that will reduce the effects of the radar noise, the transfer functions of the system shown in Figure 5-12 will be developed and examined. These transfer functions will be from the bank command and radar noise inputs to the estimated lateral position and lateral velocity outputs. These four transfer functions will be developed from the corrected observer state matrix equation, which has the form shown in (5-3), and the aircraft's reduced order-discrete model state matrix equations given in (4-7) and (4-8). The form of these discrete state equations are

$$\underline{x}(k+1) = A\underline{x}(k) + B\phi(k) \tag{5-4}$$

$$y(k) = C\underline{x}(k) \tag{5-5}$$

In addition, two output equations are needed for the observer. These will be

$$\hat{y}(k) = H_1\hat{\underline{x}}(k) \tag{5-6}$$

$$\hat{\dot{y}}(k) = H_2\hat{\underline{x}}(k) \tag{5-7}$$

The matrices H_1 and H_2 are 3x1 output matrices, where all the elements are zero except the (1, 1) element of H_1 and the (2, 1) element of H_2, which are unity.

To obtain the desired transfer functions, the five difference matrix equations, (5-3) through (5-7), must be transformed into matrix

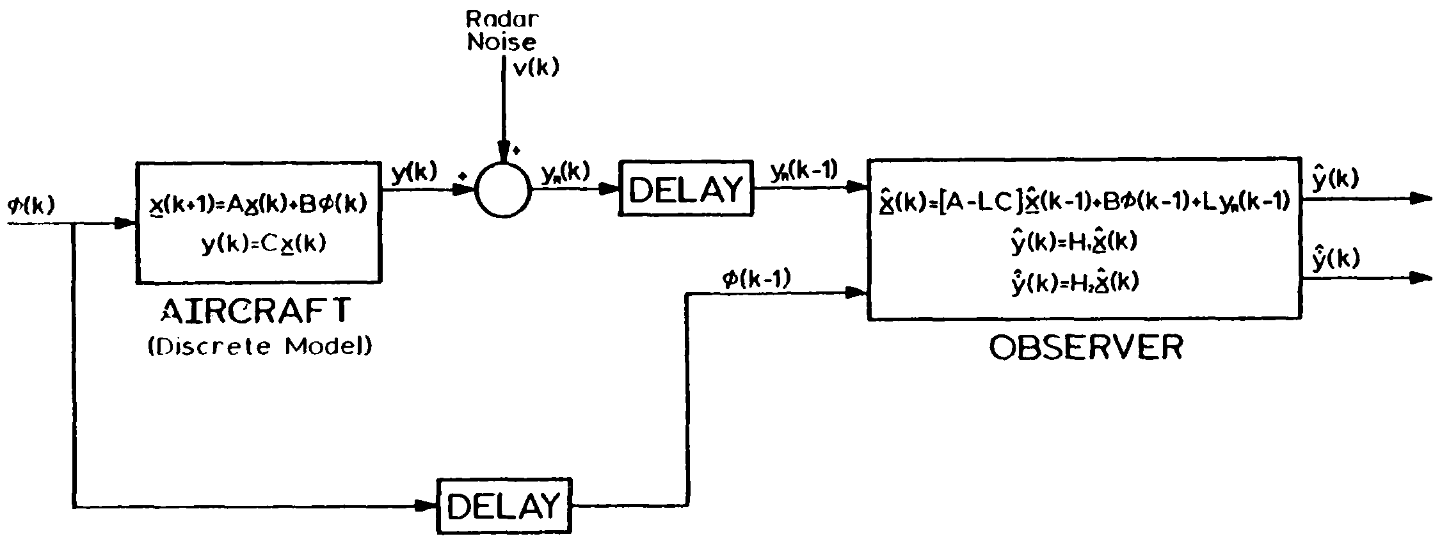


Figure 5-12. Corrected Version of the Observer.

equations in the z-domain. The z-transform method is discussed in great detail in most text books that deal with discrete systems, for example References [13] and [6]. It is important to note that after the equations have been transformed it will be possible to manipulate them using matrix algebra. The z-transform of the five difference equations are given below. These equations are arranged in the same order that the respective difference equations are presented in this section.

$$\underline{\hat{X}}(z) = [A-LC]z^{-1}\underline{\hat{X}}(z) + Bz^{-1}\Phi(z) + Lz^{-1}Y_R(z) \tag{5-8}$$

$$z\underline{X}(z) = A\underline{X}(z) + B\Phi(z) \tag{5-9}$$

$$Y(z) = C\underline{X}(z) \tag{5-10}$$

$$\hat{Y}(z) = H_1\underline{\hat{X}}(z) \tag{5-11}$$

$$\hat{\dot{Y}}(z) = H_2\underline{\hat{X}}(z) \tag{5-12}$$

The matrices in these z-domain equations are identical to their counterpart in the difference equations.

From equation (5-8), a solution for the vector $\underline{\hat{X}}(z)$ can be determined. Manipulating (5-8) gives

$$\underline{\hat{X}}(z) = [zI-A+LC]^{-1}\{B\Phi(z)+LY_R(z)\} \tag{5-13}$$

In a similar manner, a solution for the vector $\underline{X}(z)$ can be obtained

$$\underline{X}(z) = [zI-A]^{-1}B\Phi(z) \tag{5-14}$$

The matrix I in these two equations is the identity matrix, and the notation $[\cdot]^{-1}$ symbolizes the inverse matrix operation.

The process of determining the transfer functions would be simplified if an equation for $Y_R(z)$ is developed. $Y_R(z)$ is the z-transform of

the radar output signal $y_R(k)$. This signal is seen, in Figure 5-12, to be the sum of the aircraft's lateral position signal $y(k)$ and the radar noise $v(k)$. Hence an equation for $Y_R(z)$ is

$$Y_R(z) = Y(z) + V(z) \qquad (5\text{-}15)$$

A useful solution can now be developed for the vector $\hat{\underline{X}}(z)$. Manipulating equations (5-13), (5-10), (5-14), and (5-15) in the proper manner will result in

$$\hat{\underline{X}}(z) = [zI-A+LC]^{-1}\{B\Phi(z) + LC[zI-A]^{-1}B\Phi(z) + LV(z)\} \qquad (5\text{-}16)$$

Manipulating (5-16) further yields

$$\hat{\underline{X}}(z) = [zI-A+LC]^{-1}\{(I+LC[zI-A]^{-1})B\Phi(z) + LV(z)\}$$
$$\hat{\underline{X}}(z) = [zI-A+LC]^{-1}\{([zI-A]+LC)[zI-A]^{-1}B\Phi(z) + LV(z)\}$$
$$\hat{\underline{X}}(z) = [zI-A]^{-1}B\Phi(z) + [zI-A+LC]^{-1}LV(z)$$

With this solution of $\hat{\underline{X}}(z)$ and equations (5-11) and (5-12), the four desired transfer functions can be stated in matrix form.

$$\hat{Y}(z)/\Phi(z) = H_1[zI-A]^{-1}B \qquad (5\text{-}18)$$

$$\hat{Y}(z)/V(z) = H_1[zI-A+LC]^{-1}L \qquad (5\text{-}19)$$

$$\hat{\dot{Y}}(z)/\Phi(z) = H_2[zI-A]^{-1}B \qquad (5\text{-}20)$$

$$\hat{\dot{Y}}(z)/V(z) = H_2[zI-A+LC]^{-1}L \qquad (5\text{-}21)$$

Now that the desired transfer functions have been developed, a gain matrix L can be determined that will reduce the effects of the radar noise on the estimates of the aircraft's lateral position and lateral velocity.

It should be noted from these transfer functions, that the responses of the estimates to the bank command should not change with the changing of the gain matrix.

Results Using Pole Selection Technique

A single attempt was made to use the technique described above, to design an observer for use in the F4J aircraft lateral control system. In this attempt, a gain matrix was chosen that should completely eliminate the effects of the radar noise from the estimate of the aircraft's lateral velocity. This gain matrix is

$$L = \begin{bmatrix} 0.2 \\ 0.0 \\ 0.0 \end{bmatrix} \tag{5-22}$$

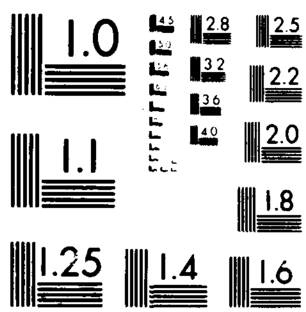
The last two elements of this matrix are chosen to be zero to force the transfer function, from radar noise input to estimated lateral position output, to zero. Once these elements are selected, the first element can be selected. This element was chosen to be 0.2 in an attempt to make the observer's dynamics "faster" than the reduced order system's dynamics. An observer was built with this gain matrix and placed in the lateral control system.

To determine the performance of this new observer, a Monte Carlo run was made. In this Monte Carlo run both the wind and radar noise [illegible] turbances were included. Unfortunately, the results [illegible] run showed that this new observer performed [illegible] filter. These results are [illegible] tion error, and [illegible]

AD-A107 384 AUBURN UNIV AL DEPT OF ELECTRICAL ENGINEERING F/G 17/7
MARINE AIR TRAFFIC CONTROL AND LANDING SYSTEM (MATCALS) INVESTI--ETC(U)
APR 81 E R GRAF, C L PHILLIPS, S A STARKS N00039-80-C-0032
UNCLASSIFIED NL

2 OF 2
AD
A107384

END
DATE
FILMED
12-81
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

A possible explanation for the poor performance of the new observer can be obtained from an examination of the pole locations of this observer. The location of this observer's poles are determined to be 0.8, 1.0, 0.867429 in the z-plane. Comparing these pole locations to the reduced order system's pole locations, given in Chapber IV, shows that the two sets of pole locations are very similar. Therefore, the dynamics of the observer, with the gain matrix given in (5-22), might not be "fast" enough to reduce the error.

VI. CONCLUSION

An observer was designed and implemented in the simulation of the F4J aircraft lateral control system. The results obtained from this implementation, given in Chapter V, demonstrate that it is possible to improve this control system's performance through the use of an observer. The vertical control system of the F4J aircraft is structurally identical to the lateral control system, and the F4J aircraft control system is typical of the aircraft control systems of the MATCALS control system. Hence, it can be concluded that the performance of the MATCALS control system may possibly be improved through the use of observers.

Further improvements in the performance of the F4J aircraft lateral control system may be achieved by using a more refined observer design. The observer design can be refined in two ways. First a higher order system can be used in the observer design process which represents the aircraft's lateral dynamics better than the third order system used in Chapter IV. The other refinement is that better pole locations can be selected for the observer. This selection can be made by trial and error or by the technique described in Chapter V.

REFERENCES

[1] David G. Luenberger, "An Introduction to Observers," IEEE Trans. Automat. Contr., Vol. AC-16, pp. 596-602, Dec. 1971.

[2] David G. Luenberger, "Observing the State of a Linear System," IEEE Trans. Mil. Electron., Vol. MIL-8, pp. 74-80, Apr. 1964.

[3] David G. Luenberger, "Observers for Multivariable Systems," IEEE Trans. Automat. Contr., Vol. AC-11, pp. 190-197, Apr. 1966.

[4] Edison Tse and Michael Athans, "Optimal Minimal-Order Observer-Estimators for Discrete Linear Time-Varying Systems," IEEE Trans. Automat. Contr., Vol. AC-12, pp. 690-698, Dec. 1967.

[5] Edison Tse, "Observer-Estimators for Discrete-Time Systems," IEEE Trans. Automat. Contr., Vol. AC-18, pp. 10-16, Feb. 1973.

[6] Gene F. Franklin and J. David Powell, Digital Control of Dynamic Systems, Addison-Wesley Publishing Company, Reading, Massachusetts, 1980.

[7] Paul M. DeRusso, Rob J. Roy, and Charles M. Close, State Variables for Engineers, John Wiley and Sons, Inc., New York, New York, 1965.

[8] Shih-Ho Wang, Edward J. Davidson, and Peter Dorato, "Observing the States of Systems with Unmeasurable Disturbances," IEEE Trans. Automat. Contr., Vol. AC-20, pp. 716-716, Oct. 1975.

[9] Charles L. Phillips, Edward R. Graf, and H. Troy Nagle, Jr., "Marine Air Traffic Control and Landing System Error and Stability Analysis," Vol. 1 and 2, Contract N00228-75-C-7080, Auburn University, Auburn University, AL, 1975.

[10] "MATCALS-AN/TPN-22 Mode 1 Final Report," ITT Gilfillan Technical Report, prepared for Naval Electronics Systems Command, Contract N00039-75-C-0021, Aug. 1979.

[11] James L. Melsa, Computer Programs for Computational Assistance, McGraw-Hill Book Company, New York, New York, 1970.

[12] Edward R. Graf, Scott A. Starks, Charles L. Phillips, and Robert W. Simpson, "Marine Air Traffic Control and Landing System Control, Radar, and Software Analysis," Contract N00228-78-C-2233, Auburn University, Auburn University, AL, 1978.

[13] Charles L. Phillips, EE 682 Class Notes, Auburn University, Auburn University, AL. To be published.

APPENDICES

APPENDIX A

DEVELOPMENT

OF

ACKERMANN'S FORMULA

The development of Ackermann's Formula for an n^{th} order system is given here. To simplify this development, the n^{th} order system described in (A-1) and (A-2) will be transformed to observer canonical form

$$\underline{x}(k+1) = A\underline{x}(k) + Bu(k) \tag{A-1}$$

$$y(k) = C\underline{x}(k) \tag{A-2}$$

A system is in observer canonical form when all the feed back loops come from the observed (output) signal; the structure of a system in this form is shown in Figure A-1. The reason for using the canonical form is that both the systems state matrix equations and the systems transfer function can be determined by inspection. This ease of determining these equations is seen in the case of the observer canonical form, where the systems state matrix equations are

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ \vdots \\ x_{n-1}(k+1) \\ x_n(k+1) \end{bmatrix} = \begin{bmatrix} a_1 & 1 & 0 & \dots & 0 \\ a_2 & 0 & 1 & \dots & 0 \\ a_3 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ a_{n-1} & 0 & 0 & \dots & 1 \\ a_n & 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ \vdots \\ x_{n-1}(k) \\ x_n(k) \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_{n-1} \\ b_n \end{bmatrix} u(k) \tag{A-3}$$

$$y(k) = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ \vdots \\ x_{n-1}(k) \\ x_n(k) \end{bmatrix} \tag{A-4}$$

and the transfer function is

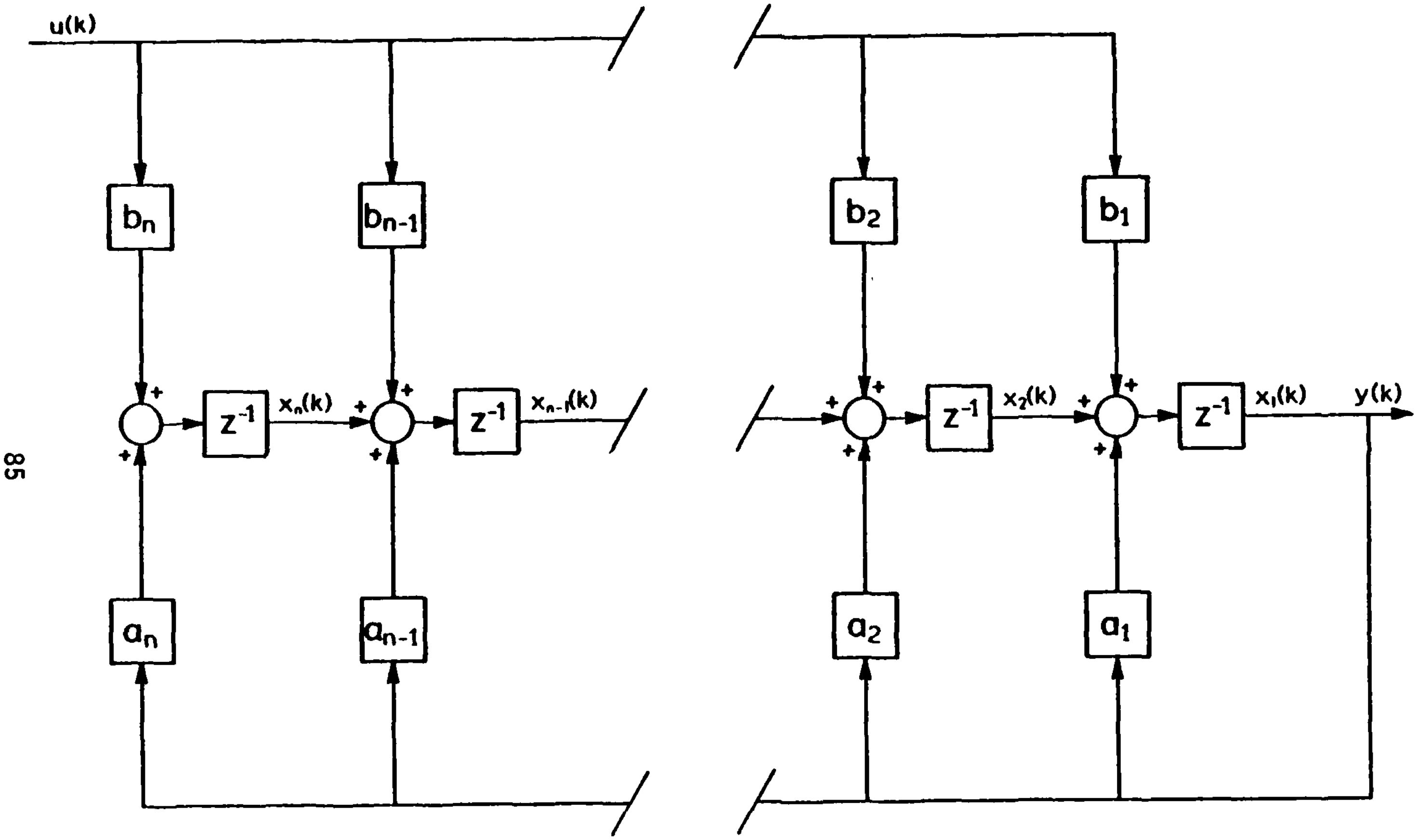


Figure A-1. A System in Observer Canonical Form.

$$G(z) = \frac{b_1 z^{n-1} + b_2 z^{n-2} + \ldots + b_{n-1} z + b_n}{z^n - a_1 z^{n-1} - a_2 z^{n-2} - \ldots - a_{n-1} z - a_n} \tag{A-5}$$

It should be noted that the transfer function can be determined directly from the state equations, with no calculations required.

Consider now the effects of an arbitrary transformation of the state vector $\underline{x}(k)$ to $\underline{w}(k)$

$$\underline{x}(k) = T\underline{w}(k) \tag{A-6}$$

where the nonsingular matrix T is nxn, and the new state vector $\underline{w}(k)$ and the old state vector $\underline{x}(k)$ are both nx1. From (A-6), a new set of system state equations can be formed.

$$\underline{x}(k+1) = A\underline{x}(k) + Bu(k) \tag{A-1}$$

$$T\underline{w}(k+1) = AT\underline{w}(k) + Bu(k)$$

$$\underline{w}(k+1) = T^{-1}AT\underline{w}(k) + T^{-1}Bu(k) \tag{A-7}$$

and

$$y(k) = C\underline{x}(k) \tag{A-2}$$

$$y(k) = CT\underline{w}(k) \tag{A-8}$$

The system described by (A-7) and (A-8) can be made to fit the observer canonical form of (A-3) and (A-4) if the correct choice of the matrix T is made. For notational purposes, if (A-7) and (A-8) do in fact describe a system that is in observer canonical form, they will be rewritten as

$$\underline{w}(k+1) = A_c\underline{w}(k) + B_c u(k) \tag{A-9}$$

$$y(k) = C_c\underline{w}(k) \tag{A-10}$$

where

$$A_c = T^{-1}AT \tag{A-11a}$$

$$B_c = T^{-1}B \tag{A-11b}$$

$$C_c = CT \tag{A-11c}$$

The subscript c will signify that the matrix is in the canonical form.

Before continuing, it is necessary to show the relationship between the observability matrix of the system described by (A-1) and (A-2) and the observability matrix of the system described by (A-9) and (A-10). The matrices are

$$\theta = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix} \tag{A-12}$$

and

$$\theta_c = \begin{bmatrix} C_c \\ C_cA_c \\ C_cA_c{}^2 \\ \vdots \\ C_cA_c{}^{n-1} \end{bmatrix} \tag{A-13}$$

respectively. Substituting (A-11) into (A-13) gives

$$\theta_c = \begin{bmatrix} CT \\ CT(T^{-1}AT) \\ CT(T^{-1}AT)^2 \\ \vdots \\ CT(T^{-1}AT)^{n-1} \end{bmatrix} = \begin{bmatrix} CT \\ CAT \\ CA^2T \\ \vdots \\ CA^{n-1}T \end{bmatrix} \tag{A-14}$$

Comparing (A-14) and (A-12) gives the desired relationship

$$\theta_c = \theta T \tag{A-15}$$

solving for T yields

$$T = \theta^{-1}\theta_c \tag{A-16}$$

Later on this will be a useful expression for the matrix T.

An observer can be designed for the system described by (A-1) and (A-2), with the observer state equations given by

$$\underline{\hat{x}}(k+1) = A\underline{\hat{x}}(k) + Bu(k) + Ly(k) - LC\underline{\hat{x}}(k) \tag{A-17}$$

Again consider a transformation of the state vector

$$\underline{\hat{x}}(k) = T\underline{\hat{w}}(k) \tag{A-18}$$

Substituting (A-18) into (A-17) gives

$$T\underline{\hat{w}}(k+1) = AT\underline{\hat{w}}(k) + Bu(k) + Ly(k) - LCT\underline{\hat{w}}(k)$$

$$\underline{\hat{w}}(k+1) = T^{-1}AT\underline{\hat{w}}(k) + T^{-1}Bu(k) + T^{-1}Ly(k) - T^{-1}LCT\underline{\hat{w}}(k) \tag{A-19}$$

Using the notation given in (A-11), (A-19) becomes

$$\underline{\hat{w}}(k+1) = A_c\underline{\hat{w}}(k) + B_c u(k) + L_c y(k) - L_c C_c\underline{\hat{w}}(k) \tag{A-20}$$

where

$$L_c = T^{-1}L \tag{A-21}$$

Now the problem is to develop an expression for the gain matrix L_c. From this, an expression for the desired gain matrix L ca. be found through (A-22).

$$L = TL_c \tag{A-22}$$

An expression for the matrix L_c can be found by matching the coefficients of the observer's characteristic polynomial to the coefficients of the desired characteristic polynomial. The characteristic polynomial of the observer, in terms of the matrix L_c, is found by

$$\alpha_L(z) = |zI-[A_c-L_cC_c]| \tag{A-23}$$

But, the matrix $[A_c-L_cC_c]$ will be in the observer canonical form, so therefore, the characteristic polynomial can immediately be written

$$\alpha_L(z) = z^n-(a_1-\ell_{c1})z^{n-1} - (a_2-\ell_{c2})z^{n-2} -\dots- (a_{n-1}-\ell_{c(n-1)})z - (a_n-\ell_{cn}) \tag{A-24}$$

If the desired characteristic polynomial of the observer is

$$\alpha(z) = z^n-\alpha_1 z^{n-1} - \alpha_2 z^{n-2} -\dots- \alpha_{n-1}z-\alpha_n \tag{A-25}$$

then the matching of the coefficients of (A-24) and (A-25) gives

$$\alpha_1 = a_1-\ell_{c1}\ , \ \alpha_2 = a_2-\ell_{c2},\ \dots,\ \alpha_{n-1} = a_{n-1}-\ell_{c(n-1)},\ \alpha_n = \alpha_n-\ell_{cn}$$

In general terms,

$$\underline{\alpha} = \underline{a} - L_c \tag{A-26}$$

Solving for the matrix L_c,

$$L_c = \underline{a} - \underline{\alpha} \tag{A-27}$$

where $\underline{a}$ is an nx1 column vector of the coefficients for the system's characteristic polynomial, and $\underline{\alpha}$ is an nx1 column vector of coefficients for the observer's desired characteristic polynomial.

A relationship between these polynomial coefficients and the system matrix A_c, is now necessary. This is done through the use of the Cayley-

Hamilton Theorem [7]. This theorem states that a matrix satisfies its own characteristic equation. For the system matrix A_c, this theorem gives

$$A_c^n - a_1 A_c^{n-1} - a_2 A_c^{n-2} - \dots - a_{n-1} A_c - a_n I = 0 \tag{A-28}$$

Next the polynomial $\alpha(A_c)$ is formed, which is the observer's desired characteristic polynomial with the matrix A_c substituted for the variable z.

$$\alpha(A_c) = A_c^n - \alpha_1 A_c^{n-1} - \alpha_2 A_c^{n-2} - \dots - \alpha_{n-1} A_c - \alpha_n I \tag{A-29}$$

Solving (A-28) for A_c^n and substituting it into (A-29) will give the relationship required.

$$\alpha(A_c) = (a_1-\alpha_1)\, A_c^{n-1} + (a_2-\alpha_2) A_c^{n-2} + \dots + (a_{n-1}-\alpha_{n-1}) A_c + (a_n-\alpha_n) I \tag{A-30}$$

At this point it is essential that the unit vector, $\underline{e}_i^n$, be defined. Let $\underline{e}_i^n$ be an nx1 column vector which is equal to the ith column of the nxn identity matrix.

Since the matrix A_c is in observer canonical form, an interesting thing happens when the unit vector $\underline{e}_n^n$ is premultiplied by the matrix A_c,

$$A_c \underline{e}_n^n = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ 0 \end{bmatrix} = \underline{e}_{n-1}^n \tag{A-31}$$

and premultiplying (A-31) by the matrix A_c again gives

$$A_c(A_c\underline{e}_n^n) = A_c^2\underline{e}_n^n = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \\ 0 \\ 0 \end{bmatrix} = \underline{e}_{n-2}^n \tag{A-32}$$

continuing in this manner will generate successive unit vectors, until

$$A_c^{n-1}\underline{e}_n^n = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \underline{e}_1^n \tag{A-33}$$

Therefore, if (A-30) is postmultiplied by $\underline{e}_n^n$, the following polynomial is obtained

$$\begin{aligned} \alpha(A_c)\underline{e}_n^n = {} & (a_1-\alpha_1)\underline{e}_1^n + (a_2-\alpha_2)\underline{e}_2^n + \ldots + (a_{n-1}-\alpha_{n-1})\underline{e}_n^{n-1} \\ & + (a_n-\alpha_n)\underline{e}_n^n \end{aligned} \tag{A-34}$$

Examination of (A-34), with the help of the relationship shown in (A-27), the following becomes apparent.

$$\alpha(A_c)\underline{e}_n^n = L_c \tag{A-35}$$

which is the needed expression of the gain matrix L_c.

With (A-35) and the relationship given in (A-22), an expression for the gain matrix L can now be found

$$\begin{aligned} L &= TL_c \qquad\qquad \text{(A-22)} \\ &= T\alpha(A_c)\underline{e}_n^n \\ &= T\alpha(T^{-1}AT)\underline{e}_n^n \\ &= TT^{-1}\alpha(A)T\underline{e}_n^n \end{aligned}$$

$$L = \alpha(A)T\underline{e}_n^n \quad \text{(A-36)}$$

The expression shown in (A-36) can be used to calculate the gain matrix L if the transformation matrix T has been found. But calculating the T matrix is not necessary if the expression developed for the T matrix in (A-16) is substituted into (A-36). Doing this gives

$$L = \alpha(A)\theta^{-1}\theta_c\underline{e}_n^n \quad \text{(A-37)}$$

It should be now noted that the product of any nxn matrix, R, and the unit vector, $\underline{e}_i^n$, will give the ith column of the matrix R. Therefore, the last two terms of (A-37) will give the nth column of the observability matrix θ_c. But if the matrix θ_c is completed, it will be found that its nth column is again the unit vector $\underline{e}_n^n$, so therefore

$$\theta_c\underline{e}_n^n = \underline{e}_n^n \quad \text{(A-38)}$$

Substituting (A-38) into (A-37) gives

$$L = \alpha(A)\theta^{-1}\underline{e}_n^n \quad \text{(A-39)}$$

which is recognized to be Ackermann's Formula as stated in (2-27).

APPENDIX B

BASIC PROGRAM

BASED ON

ACKERMANN'S FORMULA

```
1    !This program computes the gain matrix L, based on Ackermann's
2    !Formula.
10   DIM A[10,10],B[10,4],C[1,10]
20   DIM F[10,10],L[10,1]
30   DIM W[10,10],R[10,10],T[10,10],Q[10,10]
40   DIM U[1,10],H[10,1],P[10],S[1,11]
100  READ N,N1
101  !N is the order of the system and N1 is the number of inputs.
110  MAT A=ZER[N,N]
120  MAT B=ZER[N,N1]
130  MAT C=ZER[1,N]
140  MAT F=ZER[N,N]
150  MAT L=ZER[N,1]
160  MAT W=ZER[N,N]
170  MAT R=IDN[N,N]
180  MAT T=ZER[N,N]
190  MAT Q=ZER[N,N]
200  MAT U=ZER[1,N]
210  MAT H=ZER[N,1]
220  MAT P=ZER[N]
300  N2=N+1
310  MAT S=ZER[1,N2]
320  MAT  READ A,B,C,S
321  !The matrix A is the system matrix
322  !The matrix B is the input matrix
323  !The matrix C is the output matrix
324  !The matrix S contains the coefficients of the desired
325  !characteristc polynomial.
330  FOR I=1 TO N
340  J=I+1
350  P[I]=(-1)*S[1,J]
360  NEXT I
370  PRINT "MATRICES OF THE SYSTEM"
380  PRINT "MATRIX A"
390  MAT  PRINT A
400  PRINT "MATRIX B"
410  MAT  PRINT B
420  PRINT "MATRIX C"
430  MAT  PRINT C
440  PRINT "COEFFICIANTS OF DESIRED CHARACTERISTIC POLYNOMIAL"
450  MAT  PRINT S;
460  FOR I=1 TO N
470  M=N-(I-1)
480  MAT U=C*R
490  X=P[M]
500  MAT W=(X)*R
510  MAT T=T+W
520  FOR K=1 TO N
530  Q[I,K]=U[1,K]
540  NEXT K
550  MAT W=R*A
```

```
560  MAT R=W
570  NEXT I
580  MAT W=R
590  MAT T=W-T
600  MAT W=INV(Q)
610  H[N,1]=1
620  MAT Q=T*W
630  MAT L=Q*H
640  MAT W=L*C
650  MAT F=A-W
660  PRINT "MATRICES OF THE OBSERVER"
670  PRINT "MATRIX F"
680  MAT  PRINT F
690  PRINT "MATRIX B"
700  MAT  PRINT B
710  PRINT "MATRIX L"
720  MAT  PRINT L
9999 END
```

APPENDIX C

BASIC PROGRAM

TO DETERMINE

DISCRETE EQUIVALENT

MODELS

```
1    !This program calculates the discrete system model from the
2    !continuous system matrix equation.
3    !A is the continuous system's system matrix.
4    !B is the continuous system's input matrix.
5    !R is the discrete system's system matrix.
6    !S is the discrete system's input matrix.
7    !G,H,K, and L are working matrices.
100  DIM A[10,10],B[10,10],R[10,10],S[10,10]
110  DIM K[10,10],L[10,10],G[10,10],H[10,10]
200  PRINT "INPUT T, SYSTEM ORDER, AND NUMBER OF INPUTS"
210  INPUT T,N,P
220  PRINT "INPUT NUMBER OF TERMS TO BE USED IN SERIES"
230  INPUT F
240  PRINT "INPUT CONTINUOUS A MATRIX, BY ROWS"
250  MAT  INPUT A[N,N]
260  PRINT "INPUT CONTINUOUS B MATRIX, BY ROWS"
270  MAT  INPUT B[N,P]
280  PRINT
290  PRINT  USING 190;T,N,P
300  IMAGE "T=",2D.3D3X,"SYSTEM ORDER=",3D3X,"# OF INPUTS=",3D/
310  PRINT "CONTINUOUS SYSTEM A MATRIX IS"
320  MAT  PRINT A
330  PRINT "CONTINUOUS B MATRIX IS"
340  MAT  PRINT B
350  REM TO INITIALIZE AND SET DIMENSIONS ON MATRICES
360  MAT R=IDN[N,N]
370  MAT S=IDN[N,N]
380  MAT L=IDN[N,N]
390  MAT G=ZER[N,N]
400  MAT H=ZER[N,N]
410  MAT K=ZER[N,N]
420  FOR J=1 TO F
430  MAT G=L
440  MAT L=(T/(J+1))*G
450  MAT G=L
460  MAT L=G*A
470  MAT S=S+L
480  MAT K=(J+1)*L
490  MAT R=R+K
500  NEXT J
510  MAT H=S
520  MAT S=(T)*H
530  MAT G=ZER[N,P]
540  MAT G=S*B
550  PRINT "DISCRETE A MATRIX IS"
560  MAT  PRINT R
570  PRINT "DISCRETE B MATRIX IS"
580  MAT  PRINT G
9999 END
```

APPENDIX D

SIMULATION

OF THE

F4J AIRCRAFT

LATERAL CONTROL SYSTEM

```
C     *****************************************************************
C     This program simulates the Control System of the F4J Aircraft.
C     *****************************************************************
C     This portion of the program sets up the system.
C     *****************************************************************
      COMMON/MAT/A(23,23),B(23,10),C(8,23),D(8,10)
      COMMON/VECT/X(23),U(10),FN(23)
      COMMON/CONINT/XI(23),N,NU,NY,NF
      COMMON/CONREL/T,H,SFREQ,TZERO,SR
      COMMON/RAMD/IX,IY,IZ
    4 FORMAT('0',9X,'THE STARTING RANGE IS',F20.5)
    7 FORMAT('-',9X,'N=',I2,10X,'NU=',I2,10X,'NY=',I2,10X,'NF=',I2,//)
    8 FORMAT(8F10.3)
    9 FORMAT(F20.5)
   10 FORMAT(5F10.5)
  100 FORMAT(5X,I5,5X,I5,5X,I5)
 1000 FORMAT('-',9X,'TZERO= ',F10.6,3X,'TF= ',F10.6,3X,'H=',F10.6,3X,
     &'FREQ=',F10.6,3X,'SFREQ=',F10.6,//)
 1001 FORMAT('0',10X,'THE A MATRIX'/)
 1002 FORMAT('0',10E13.4)
 1003 FORMAT('0',10X,'INITIAL CONDITIONS FOR STATE VECTOR,X'/)
 1004 FORMAT('0',10X,'THE B MATRIX'/)
 1007 FORMAT('0',10X,'THE C MATRIX'/)
 1010 FORMAT(1H1,4X,23HGRAPHICAL TIME RESPONSE)
 1011 FORMAT(//,133(1H*))
 1111 FORMAT('0',10X,'THE D MATRIX',/)
      READ(5,100) IX,IY,IZ
C     *****************************************************************
C     The values IX,IY,and IZ initialize the random number generators
C     of the wind and radar noise disturbances.
C     *****************************************************************
      CALL MATRIX
      READ(5,10) TZERO,TF,H,FREQ,SFREQ
C     *****************************************************************
C     TZERO is the starting time, TF is the final time, H is the
C     integration period, FREQ is the output frequence, and SFREQ is the
C     sampling frequence.
C     *****************************************************************
      READ(5,8) (XI(I),I=1,N)
C     *****************************************************************
C     The vector XI is the initial condition vector of the aircraft.
C     *****************************************************************
      READ(5,9) SR
C     *****************************************************************
C     SR is the starting range.
C     *****************************************************************
      WRITE(6,1010)
      WRITE(6,1011)
      WRITE(6,1000) TZERO,TF,H,FREQ,SFREQ
      WRITE(6,7) N,NU,NY,NF
```

```
      WRITE(6,1001)
      DO 2 I=1,N
    2 WRITE(6,1002) (A(I,J),J=1,N)
      WRITE(6,1004)
      DO 62 I=1,N
   62 WRITE(6,1002) (B(I,J),J=1,NU)
      WRITE(6,1007)
      DO 64 I=1,NY
   64 WRITE(6,1002) (C(I,J),J=1,N)
      WRITE(6,1111)
      DO 88 I=1,NY
   88 WRITE(6,1002) (D(I,J),J=1,NU)
      WRITE(6,4) SR
      WRITE(6,1003)
      WRITE(6,1002) (XI(I),I=1,N)
      WRITE(6,1011)
      IFQ=FREQ
      T=TZERO
      CALL TRESP(TF,IFQ)
      STOP
      END
```

```
      BLOCK DATA
C     ********************************************************************
C     This subroutine clears all the matrices and vectors.
C     ********************************************************************
      COMMON/MAT/A(23,23),B(23,10),C(8,23),D(8,10)
      COMMON/VECT/X(23),U(10),FN(23)
      DATA A/529*0.0/,B/230*0.0/,C/184*0.0/,D/80*0.0/
      DATA X/23*0.0/,U/10*0.0/,FN/23*0.0/
      END
```

```
      SUBROUTINE MATRIX
C     ******************************************************************
C     This subroutine generates the non-zero elements of the A,B and C
C     matrices.
C     ******************************************************************
      COMMON/MAT/A(23,23),B(23,10),C(8,23),D(8,10)
      COMMON/VECT/X(23),U(10),FN(23)
      COMMON/CONINT/XII(23),N,NU,NY,NF
      DATA AM,AXE,BB,CB,G,GE,RE/1057.,.2234,38.666,16.04,32.2,-.0611,
     &.002378/
      DATA S,TE,VE,W,XI,XZI,ZI/530.,.1623,220.8,34000.,25600.,21500.,
     &145200./
      DATA CLB,CLP,CLR,CLDA,CLDR,CNB,CNP,CNR/-.1565,-.275,.207,-.0573,
     &.00286,.1982,-.013,-.31/
      DATA CNDA,CNDR,CYB,CYP,CYR,CYDA,CYDR/-.0043,-.0722,-.647,1.26,
     &.717,-.0356,.1345/
      N=9
      NU=3
      NF=3
      NY=2
      B1=RE*S*VE**2/(2.*(XI*ZI-XZI**2))*BB
      A(1,4)=1.
      A(1,5)=TAN(TE)
      A(2,5)=1./COS(TE)
      A(3,1)=G*COS(TE)/VE
      A(3,3)=RE*S*VE*CYB/(2.*AM)
      A(3,4)=RE*S*BB*CYP/(4.*AM)+SIN(AXE)
      A(3,5)=RE*S*BB*CYR/(4.*AM)-COS(AXE)
      A(3,7)=RE*S*VE*CYDA/(2.*AM)/57.3
      A(3,9)=19.36*RE*S*VE*CYDR/(2.*AM)/57.3
      A(4,3)=B1*(ZI*CLB+XZI*CNB)
      A(4,4)=B1*BB/(2.*VE)*(ZI*CLP+XZI*CNP)
      A(4,5)=B1*BB/(2.*VE)*(ZI*CLR+XZI*CNR)
      A(4,7)=B1*(ZI*CLDA+XZI*CNDA)/57.3
      A(4,9)=19.36*B1*(ZI*CLDR+XZI*CNDR)/57.3
      A(5,3)=B1*(XZI*CLB+XI*CNB)
      A(5,4)=B1*BB/(2.*VE)*(XZI*CLP+XI*CNP)
      A(5,5)=B1*BB/(2.*VE)*(XZI*CLR+XI*CNR)
      A(5,7)=B1*(XZI*CLDA+XI*CNDA)/57.3
      A(5,9)=19.36*B1*(XZI*CLDR+XI*CNDR)/57.3
      A(6,1)=-VE*SIN(AXE)
      A(6,2)=VE*COS(GE)
      A(6,3)=VE
      A(7,7)=-10.
      A(8,4)=.0262*57.3
      A(8,5)=.9996*57.3
      A(8,8)=-.5
      A(9,9)=-20.
      B(3,2)=RE*S*VE*CYB/(2.*AM)
      B(4,2)=B1*(ZI*CLB+XZI*CNB)
```

```
B(5,2)=B1*(XZI*CLB+XI*CNB)
C(1,6)=1.0
RETURN
END
```

```
      SUBROUTINE TRESP(TF,IFREQ)
C     ******************************************************************
C     This subroutine gives the time response of the control system.
C     ******************************************************************
      COMMON/MAT/A(23,23),B(23,10),C(8,23),D(8,10)
      COMMON/VECT/X(23),U(10),FN(23)
      COMMON/PUCC/P(10,100)
      COMMON/COMP/R(10,100)
      COMMON/CONINT/XI(23),N,NU,NY,NF
      COMMON/CONREL/T,H,SFREQ,TZERO,SR
      COMMON/DTT/DTSAMP
      COMMON/ESTA/EST1,EST2,EST3
      COMMON/ABFIL/ABY,ABYDOT
      COMMON/COUNT/NCON
      COMMON/MSESDS/SDEPS,SDAPS,SDEVS,SDAVS,SPOSS
      COMMON/NOISE/X6DUM
      COMMON/RAMD/IX,IY,IZ
      COMMON/MONTE/ICC,X6A(1000),X6S(1000),TM(1000)
      DIMENSION IR(10,100)
      REAL MSEP,MSEEP,MSEAP,MSEEV,MSEAV
   10 FORMAT(F6.3,',',E10.3,',',E10.3,',',E10.3,',',E10.3,',',
     &E10.3,',',E10.3)
   11 FORMAT(' ',4X,'T=',F8.3,4X,'AVG=',F11.7,4X,'RMS=',F11.7)
   12 FORMAT(///,10X,'TOTAL',5X,'AVG',F11.7,4X,'RMS',F11.7)
   13 FORMAT(/,2X,I3,2X,E14.5,2X,E14.5,2X,E14.5,2X,I10,1X,I10,1X,I10)
   15 FORMAT(//,2X,'RUN',4X,'RMSE OF POS.',6X,'FINAL POS.',6X,'MAX OFFSE
     &T',8X,'IX',9X,'IY',9X,'IZ')
   21 FORMAT('+',23X,E14.5)
   22 FORMAT(//,5X,'RMSE OF TRUE POSITION')
   23 FORMAT(//,5X,'POSITION')
   24 FORMAT(//,5X,'VELOCITY')
   26 FORMAT(/,5X,'RMSE OF ESTIMATOR=',20X,'RMSE OF ABFILTER=')
   27 FORMAT('+',19X,E14.5,23X,E14.5)
   28 FORMAT('1',5X,'T',8X,'TTD',10X,'X(6)',9X,'X6(DX)',9X,'ESTY',
     &11X,'ABY',10X,'FN(6)',9X,'ESTYD',9X,'ABYD')
   29 FORMAT(' ',F8.3,2X,F8.3,2X,8(1PE14.5))
 1000 FORMAT(/,5X,33HMAXIMUM NUMBER OF POINTS EXCEEDED /)
      NR=20
      NR=1
C     ******************************************************************
C     NR is the number of simulation runs to be used in a Monte Carlo
C     run.
C     ******************************************************************
      NT=TF*SFREQ +1.0
      DO 3 I=1,NT
      X6A(I)=0.0
    3 X6S(I)=0.0
      DO 1 INR=1,NR
      IR(1,INR)=IX
      IR(2,INR)=IY
```

```
      IR(3,INR)=IZ
      DO 2 I=1,N
      U(I)=0.0
    2 X(I)=XI(I)
      T=TZERO
      ICC=0
      AMX6=0.0
      J=0
      NP=0
      NCON=1
      SPOSS=0.0
      SDEPS=0.0
      SDAPS=0.0
      SDEVS=0.0
      SDAVS=0.0
      OUTT=0.
      DIGT=0.
      PDT=H*IFREQ
      CALL DIGFIL(DIGT)
  450 CONTINUE
      TTD=(SR-220.39*T)/220.39
      NP=NP+1
      P(1,NP)=T
      P(2,NP)=X(6)
      P(3,NP)=EST1
      P(4,NP)=ABY
      P(5,NP)=FN(6)
      P(6,NP)=EST2
      P(7,NP)=ABYDOT
      P(8,NP)=X6DUM
      P(9,NP)=TTD
      P(10,NP)=0.0
      OUTT=OUTT+PDT
      J=J+1
      IF(J.GT.101)GO TO 222
      IF(T.GE.TF) GO TO 400
   50 CONTINUE
      DO 14 II=1,4
      CALL AXDOT
   14 CALL RUNGE(II)
      AX6=ABS(X(6))
      AMX6=AMAX1(AMX6,AX6)
      IF(ICC.GE.NT) GOTO 400
      IF((T*1.001).GE.DIGT) CALL DIGFIL(DIGT)
      IF((T*1.001).GE.OUTT) GO TO 450
      IF(T.GE.TF) GO TO 400
      GO TO 50
  222 WRITE(6,1000)
  400 CONTINUE
      MSEP=SQRT(SPOSS/NCON)
```

```
      MSEEP=SQRT(SDEPS/NCON)
      MSEAP=SQRT(SDAPS/NCON)
      MSEEV=SQRT(SDEVS/NCON)
      MSEAV=SQRT(SDAVS/NCON)
      R(1,INR)=MSEP
      R(2,INR)=MSEEP
      R(3,INR)=MSEAP
      R(4,INR)=MSEEV
      R(5,INR)=MSEAV
      R(6,INR)=P(2,NP)
      R(7,INR)=AMX6
      IX=IX+3
      IY=IY+3
      IZ=IZ+3
    1 CONTINUE
      DO 4 I=1,NT
      X6A(I)=X6A(I)/NR
    4 X6S(I)=SQRT(X6S(I)/NR)
      X6AA=0.0
      X6SA=0.0
      DO 6 I=1,NT
      X6AA=X6AA+X6A(I)
    6 X6SA=X6SA+X6S(I)
      X6AA=X6AA/NT
      X6SA=X6SA/NT
      PRINT 12,X6AA,X6SA
      WRITE(6,15)
      DO 7 I=1,NR
    7 WRITE(6,13) I,R(1,I),R(6,I),R(7,I),IR(1,I),IR(2,I),IR(3,I)
      NI=0
      WRITE(6,28)
      DO 101 I=1,NP
  101 WRITE(6,29) P(1,I),P(9,I),P(2,I),P(8,I),P(3,I),P(4,I),P(5,I),
     &P(6,I),P(7,I)
      WRITE(6,22)
      WRITE(6,21) MSEP
      WRITE(6,23)
      WRITE(6,26)
      WRITE(6,27)MSEEP,MSEAP
      WRITE(6,24)
      WRITE(6,26)
      WRITE(6,27)MSEEV,MSEAV
      DO 100 I=1,NP
  100 PUNCH 10,P(1,I),P(2,I),P(3,I),P(4,I),P(5,I),P(6,I),P(7,I)
      IF(NI.EQ.0) GOTO 401
      DO 5 I=1,NT
    5 PRINT 11,TM(I),X6A(I),X6S(I)
  401 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DIGFIL(DIGT)
C     ******************************************************************
C     This subroutine simulates the digital controlling unit.
C     ******************************************************************
      IMPLICIT REAL(K)
      COMMON/MAT/A(23,23),B(23,10),C(8,23),D(8,10)
      COMMON/VECT/X(23),U(10),FN(23)
      COMMON/CONINT/XI(23),N,NU,NY,NF
      COMMON/CONREL/T,H,SFREQ,TZERO,SR
      COMMON/GAUSSC/SIGMA,XMEAN
      COMMON/DTT/DTSAMP
      COMMON/RANG/RANGE
      COMMON/ABFIL/ABY,ABYDOT
      COMMON/COUNT/NCON
      COMMON/MSED/DEP,DAP,DEV,DAV,POS
      COMMON/MSEDS/DEPS,DAPS,DEVS,DAVS,POSS
      COMMON/MSESDS/SDEPS,SDAPS,SDEVS,SDAVS,SPOSS
      COMMON/ESTA/EST1,EST2,EST3
      COMMON/NOISE/X6DUM
      COMMON/MONTE/ICC,X6A(1000),X6S(1000),TM(1000)
      DIMENSION DT(5),KRL(5),A1L(5),A2L(5),A3L(5),
     &TRL(5),KCL(5),TAL(5),TIL(5),ALPHAB(5),KBC1(5),KBC3(5),KBC2(5),
     &KBC4(5),KBC5(5),A4L(5)
      DATA K7/8./
      DATA S1,S2,S3,S4,S5,S6,S7,S8,S9,S10,S11/11*0.0/
      DATA DT/.2,.1,.05,.033333,.025/
      DATA KRL/.534,.3174,.1738,.1195,.091/
      DATA A1L/.586,.766,.875,.915,.935/
      DATA A2L/.414,.234,.125,.085,.065/
      DATA A3L/.414,.234,.125,.085,.065/
      DATA A4L/.024,.012,.0061,.0041,.00306/
      DATA TRL/5*7.5/
      DATA TAL/5*7.5/
      DATA KCL/5*.1/
      DATA TIL/5*30./
      DATA ALPHAB/.2275,.1211,.0625,.0421,.03175/
      DATA KBC1/.96667,.98113,.99020,.99341,.99504/
      DATA KBC2/1.4,1.81132,1.94118,1.96932,1.98145/
      DATA KBC3/.96667,.98113,.99020,.99341,.99504/
      DATA KBC4/1.4,1.81132,1.94118,1.96932,1.98145/
      DATA KBC5/.9333,.96226,.98039,.98681,.99007/
      DATA K3/.57735/,XDOT/220.39/
      IF(T .NE. TZERO) GOTO 111
      S1=0.0
      S2=0.0
      S3=0.0
      S4=0.0
      S5=0.0
      S6=0.0
      S7=0.0
```

```
      S8=0.0
      S9=0.0
      S10=0.0
      S11=0.0
  111 CONTINUE
      DTSAMP=1./SFREQ
      IF (DTSAMP.GT.0.15) I=1
      IF (DTSAMP.LE.0.15.AND.DTSAMP.GT.0.075) I=2
      IF (DTSAMP.LE.0.07500.AND.DTSAMP.GT.0.04167) I=3
      IF (DTSAMP.LE.0.04167.AND.DTSAMP.GT.0.02911) I=4
      IF (DTSAMP.LE.0.02911) I=5
      RANGE = SR-T*220.39
      IF(RANGE.GE.5000.) K1L=5000./RANGE
      IF(RANGE.GE.50000.) K1L=.1
      IF(RANGE.LT.5000.) K1L=1.
      K2L=1.0
      K3L=1.0
      K4L=K1L
      X6DUM=X(6)
      X6DUM=X6(DX)
C     ******************************************************************
C     Depending on which X6DUM statement is used determines if the radar
C     noise is included in the simulation run.
C     ******************************************************************
      CALL ESTMAT(X6DUM,I)
      CALL ABFLTR(X6DUM,I)
      YIN=X(6)
      YDOTIN=FN(6)
      YIN=ABY
      YDOTIN=ABYDOT
      YIN=EST1
      YDOTIN=EST2
C     ******************************************************************
C     Depending on which set of YIN and YDOTIN statements are used
C     determines which values are used for the estimates of the lateral
C     position and lateral velocity. These are, the true values, the
C     A-B Filter estimates, or the Observer estimates.
C     ******************************************************************
      Y=-YIN
      YDOT=-YDOTIN
      IF(RANGE.GE.16000.) K4L=0.
      IF(ABS(Y).GE.100.) K4L=0.
      YDOTF=S3+KRL(I)*(YDOT-S3)
      Y2DOTP=S4*A1L(I)+A2L(I)/DT(I)*(YDOTF-S3)
      Y2DOT=S5+A3L(I)*(Y2DOTP-S5)
      PHIINT=S6+K4L*KCL(I)/TIL(I)*(Y+S1)/2.*DT(I)
      IF(PHIINT.GT.10.) PHIINT=10.
      IF(PHIINT.LT.-10.) PHIINT=-10.
      CTLC=ABS(K3*K2L*XDOT*TRL(I))
      IF(ABS(Y).GE.CTLC) Y=SIGN(CTLC,Y)
```

```
      PHIC4P=K1L*KCL(I)*(Y+K2L*TRL(I)*YDOTF+K3L*TAL(I)*Y2DOT)
      IF(RANGE.GT.5000.) GO TO 1001
      KVL=5.+.0032*RANGE
      IF(RANGE.LT.0.) KVL=5.
      PHIC3P=PHIC4P
      IF(ABS(PHIC4P).GT.KVL) PHIC3P=SIGN(KVL,PHIC4P)
      GO TO 1002
 1001 PHIC3P=PHIC4P
      IF(ABS(PHIC4P).GT.21.) PHIC3P=SIGN(21.,PHIC4P)
 1002 CONTINUE
      PHIC2P=PHIC3P+PHIINT
      PHIC1P=S7+ALPHAB(I)*(PHIC2P-S7)
      PHICX=KBC1(I)*PHIC1P-KBC2(I)*S7+KBC3(I)*S8+KBC4(I)*S9-KBC5(I)*S10
      PHICX=PHIC1P
      PHICL=A4L(I)*PHICX+(1.-A4L(I))*S11
      S1=Y
      S2=YDOT
      S3=YDOTF
      S4=Y2DOTP
      S5=Y2DOT
      S6=PHIINT
      S8=S7
      S7=PHIC1P
      S10=S9
      S9=PHICX
      S11=PHICL
      IF((PHICX-PHICL).GT.K7) GO TO 10
      IF((PHICX-PHICL).LT.-K7) GO TO 11
      U(1)=PHICX
      GO TO 12
   10 U(1)=PHICL+K7
      GO TO 12
   11 U(1)=PHICL-K7
   12 CONTINUE
      IF(U(1).LT.-30.) U(1)=-30.
      IF(U(1).GT.30.) U(1)=30.
      DIGT=DIGT+DTSAMP
  764 CONTINUE
      ICC=ICC+1
      X6A(ICC)=X6A(ICC)+X(6)
      X6S(ICC)=X6S(ICC)+X(6)**2
      TM(ICC)=T
      NCON=NCON+1
      POS=X(6)
      DEP=X(6)-EST1
      DAP=X(6)-ABY
      DEV=FN(6)-EST2
      DAV=FN(6)-ABYDOT
      POSS=POS**2
      DEPS=DEP**2
```

```
DAPS=DAP**2
DEVS=DEV**2
DAVS=DAV**2
SPOSS=SPOSS+POSS
SDEPS=SDEPS+DEPS
SDAPS=SDAPS+DAPS
SDEVS=SDEVS+DEVS
SDAVS=SDAVS+DAVS
RETURN
END
```

```
      SUBROUTINE RUNGE(II)
C     ******************************************************************
C     This subroutine performs the integration on the continuous
C     aircraft system.
C     ******************************************************************
      COMMON/MAT/A(23,23),B(23,10),C(8,23),D(8,10)
      COMMON/VECT/X(23),U(10),FN(23)
      COMMON/CONREL/T,H,SFREQ,TZERO,SR
      COMMON/CONINT/XI(23),N,NU,NY,NF
      DIMENSION SAVEX(11),PHI(11)
      GO TO (12,13,4,15),II
   12 H1=.5*H
      DO 600 J=1,N
      SAVEX(J)=X(J)
      PHI(J)=FN(J)
  600 X(J)=SAVEX(J)+H1*FN(J)
      T=T+H1
      RETURN
   13 H1=.5*H
      DO 700 J=1,N
      PHI(J)=PHI(J)+2.*FN(J)
  700 X(J)=SAVEX(J)+H1*FN(J)
      RETURN
    4 DO 800 J=1,N
      PHI(J)=PHI(J)+2.*FN(J)
  800 X(J)=SAVEX(J)+H*FN(J)
      T=T+.5*H
      RETURN
   15 H2=H/6.
      DO 900 J=1,N
  900 X(J)=SAVEX(J)+H2*(PHI(J)+FN(J))
      RETURN
      END
```

```
      SUBROUTINE AXDOT
C     ******************************************************************
C     This subroutine updates the continuous system for the integration
C     procedure.
C     ******************************************************************
      COMMON/MAT/A(23,23),B(23,10),C(8,23),D(8,10)
      COMMON/VECT/X(23),U(10),XDOT(23)
      CAL(Z1,Z2)=SIGN(AMIN1(ABS(Z1),Z2),Z1)
      U(2)=0.0
      U(2)=WIND(DY)
C     ******************************************************************
C     Depending on which U(2) statement is used determines if the wind
C     distubance is included in the simulation.
C     ******************************************************************
      F1=CAL(U(1),14.)
      XDOT(1)=A(1,4)*X(4)+A(1,5)*X(5)
      XDOT(2)=A(2,5)*X(5)
      XDOT(3)=A(3,1)*X(1)+A(3,3)*X(3)+A(3,4)*X(4)+A(3,5)*X(5)+A(3,7)*
     &X(7)+A(3,9)*X(9)+B(3,2)*U(2)
      XDOT(4)=A(4,3)*X(3)+A(4,4)*X(4)+A(4,5)*X(5)+A(4,7)*X(7)+A(4,9)*
     &X(9)+B(4,2)*U(2)
      XDOT(5)=A(5,3)*X(3)+A(5,4)*X(4)+A(5,5)*X(5)+A(5,7)*X(7)+A(5,9)*
     &X(9)+B(5,2)*U(2)
      XDOT(6)=A(6,1)*X(1)+A(6,2)*X(2)+A(6,3)*X(3)
      F2=171.9*X(1)+68.76*X(4)-3.*F1
      F2=CAL(F2,7.5)
      XDOT(7)=A(7,7)*X(7)+10.*F2
      XDOT(8)=A(8,4)*X(4)+A(8,5)*X(5)+A(8,8)*X(8)
      F3=.67*F2+143.2*X(5)+3.753*X(4)
     &-1.25*X(8)+211.97*X(5)
     &+211.97*XDOT(3)
      F3=CAL(F3,5.)
      XDOT(9)=A(9,9)*X(9)+F3
      RETURN
      END
```

```
      SUBROUTINE ABFLTR(X6DUM,I)
C     *****************************************************************
C     This subroutine simulates the A-B Filter.
C     *****************************************************************
      COMMON/MAT/A(23,23),B(23,10),C(8,23),D(8,10)
      COMMON/VECT/X(23),U(10),FN(23)
      COMMON/CONREL/T,H,SFREQ,TZERO,SR
      COMMON/ABFIL/Y,YDOT
      DIMENSION DT(5),ALPHAY(5),BETAY(5)
      DATA S1,S2,S3/3*0.0/
      DATA DT/.2,.1,.05,.033333,.025/
      DATA ALPHAY/.7599,.51,.3,.2116,.1633/
      DATA BETAY/.4656,.1746,.0529,.025,.0145/
      IF(T .NE. TZERO) GOTO 111
      S1=0.0
      S2=0.0
      S3=0.0
  111 CONTINUE
      YERF=X6DUM
      YP=S1+DT(I)*S2
      Y=YP+ALPHAY(I)*(YERF-YP)
      YDOT=S2+BETAY(I)/DT(I)*(YERF-YP)
      S1=Y
      S2=YDOT
      RETURN
      END
```

```
      SUBROUTINE ESTMAT(X6DUM,IDTS)
C     ******************************************************************
C     This subroutine simulates the Observer.
C     ******************************************************************
      COMMON/MAT/A(23,23),B(23,10),C(8,23),D(8,10)
      COMMON/VECT/X(23),U(10),FN(23)
      COMMON/ESTA/Y1,Y2,Y3
      COMMON/ESTS/F(10,10),O(10,10),E(10,10),G(10,4),L(10),H(4)
      COMMON/EST1/EX(10),S(10),SU(4),SY(4),Q
      COMMON/CONREL/T,HH,SFREQ,TZERO,SR
      DIMENSION Y(10)
      REAL L,H,O
   10 FORMAT (3I2)
   20 FORMAT (4E16.7)
   30 FORMAT (/,6E20.7)
   40 FORMAT (//)
      DATA NC/1/
      IF(NC.NE.1) GOTO 901
      NC=0
      READ 10,NE,NU,NY
      DO 105 I=1,NE
  105 READ 20,(O(I,J),J=1,NE)
      DO 102 I=1,NE
  102 READ 20,(G(I,J),J=1,NU)
  104 READ 20,(H(J),J=1,NE)
      DO 103 I=1,NE
  103 READ 20,L(I)
      PRINT 40
      PRINT 10,NE,NU,NY
      PRINT 40
      DO 205 I=1,NE
  205 PRINT 30,(O(I,J),J=1,NE)
      PRINT 40
      DO 202 I=1,NE
  202 PRINT 30,(G(I,J),J=1,NU)
      PRINT 40
  204 PRINT 30,(H(J),J=1,NE)
      PRINT 40
      DO 203 I=1,NE
  203 PRINT 30,L(I)
      PRINT 40
  901 CONTINUE
      NC=0
      IF(T .NE. TZERO) GOTO 111
      DO 501 I=1,NE
  501 S(I)=0.0
      DO 502 I=1,NU
  502 SU(I)=0.0
      DO 503 I=1,NY
  503 SY(I)=0.0
```

```
111 CONTINUE
    DO 402 I=1,NU
402 SU(I)=U(I)
    DO 403 I=1,NY
403 SY(I)=X6DUM
    DO 603 I=1,NE
    DO 603 J=1,NE
603 E(I,J)=L(I)*H(J)
    DO 604 I=1,NE
    DO 604 J=1,NE
604 F(I,J)=O(I,J)-E(I,J)
    DO 300 I=1,NE
300 Y(I)=0.0
    DO 301 I=1,NE
    DO 301 J=1,NE
301 Y(I)=Y(I)+F(I,J)*S(J)
    DO 302 I=1,NE
    DO 302 J=1,NU
302 Y(I)=Y(I)+G(I,J)*SU(J)
    DO 303 I=1,NE
    DO 303 J=1,NY
303 Y(I)=Y(I)+L(I)*SY(J)
    Y1=S(1)
    Y2=S(2)
    Y3=S(3)
    DO 401 I=1,NE
401 S(I)=Y(I)
    RETURN
    END
```

```
C     *****************************************************************
C     The remainder of this program is devoted to generation of the wind
C     and the radar noise disturbances.
C     *****************************************************************


      FUNCTION WIND(DY)
      SDYWSY=0.0
      WDSCLY=0.5
      DY=1.0
      WIND=SDYWSY+WDSCLY*(RANNU2(DX)-0.5)*2
      RETURN
      END


      FUNCTION RANNUM(DX)
      COMMON/RAMD/IX,IY,IZ
      IX=IX*65539
      IF(IX) 2,2,3
    2 IX=IX+2147483647 + 1
    3 RANNUM=(FLOAT(IX) *0.4656613D-9)
      RETURN
      END


      FUNCTION RANNU1(DX)
      COMMON/RAMD/IX,IY,IZ
      IY=IY*65539
      IF(IY) 2,2,3
    2 IY=IY+2147483647 + 1
    3 RANNU1=(FLOAT(IY) *0.4656613D-9)
      RETURN
      END


      FUNCTION RANNU2(DX)
      COMMON/RAMD/IX,IY,IZ
      IZ=IZ*65539
      IF(IZ) 2,2,3
    2 IZ=IZ+2147483647+1
    3 RANNU2=(FLOAT(IZ) *0.4656613D-9)
      RETURN
      END
```

```
      FUNCTION GAUS1(DX)
      DATA SIGM1/1./,XMEA1/0./
      A1=0.0
      DO 50 I=1,12
   50 A1=A1+RANNU1(DX)
      GAUS1=(A1-6.0)*SIGM1+XMEA1
      RETURN
      END


      FUNCTION GAUSS(DX)
      DATA SIGMA/1./,XMEAN/0./
      A1=0.0
      DO 50 I=1,12
   50 A1=A1+RANNUM(DX)
      GAUSS=(A1-6.0)*SIGMA+XMEAN
      RETURN
      END


      FUNCTION X6(DX)
      COMMON/VECT/X(23),U(10),FN(23)
      COMMON/CONREL/T,H,SFREQ,TZERO,SR
      COMMON/RANG/RANGE
      DATA S1,S2,S3,S4/4*0./
      IF(T .NE. TZERO) GOTO 111
      S1=0.0
      S2=0.0
      S3=0.0
      S4=0.0
  111 CONTINUE
      PHI=ATAN((X(6)+178.1)/(RANGE+762.9))
      DPHIN=.382*S1+.15*S2+.122*S3-.0045*S4+.0005657*GAUSS(DX)
      PHIM=PHI+DPHIN+.000148*GAUS1(DX)
      S4=S3
      S3=S2
      S2=S1
      S1=DPHIN
      X6=(RANGE+762.9)*TAN(PHIM)-178.1
      RETURN
      END
```

Part Three

A CENTROID ALGORITHM BASED UPON RETURN AMPLITUDE - VERSUS-ANGLE SIGNATURE

Prepared for

Under

by

Electrical Engineering Department
Auburn University
Auburn, Alabama

Prepared by: R. J. Machuzak and E. R. Graf

Radar Centroid Investigation

A method of estimating the centroid location of a target utilizing scan return amplitude-versus-angle information was introduced. The method was compared to three thresholding estimators and a first moment estimator in a computer-simulated automatic landing system.

It was found that the method introduced was the most robust and accurate of the estimators in noise, due to its unique scan rejection capability. In periods of high signal-to-noise ratio the method had less error than the thresholding methods, and was similar in ability to the first moment estimator. Further, the pulse transmissions required to obtain a desired level of performance was much reduced from the thresholding methods employed in the simulation.

Table of Contents

I. INTRODUCTION

Essential to the performance of any tracking radar is an effective target centroid estimator. The purpose of this investigation was to examine the accuracy of several target centroid estimators in a comparative fashion, and to develope a non-thresholding algorithm as part of this project. This analysis was conducted using a simulation of a landing radar tracking a passive target.

Only the fundamental features of the new algorithm and its development are presented here. A complete description and thorough analysis are being compiled for a subsequent report.

II. OVERVIEW OF THE SIMULATION

The computer simulation describes a large jet fighter aircraft in a normal ground controlled approach (GCA) with the radar antenna located 500 meters from the runway touchdown point, as shown in Figure 2-1. The simulation initiates the flight with the target 3.72 nmi downrange from the runway touchdown point, or 4.0 nmi downrange from the radar antenna. The target model is allowed to approach the runway at a constant 148.6 mph on a 3.5 degree glideslope. The radar is a phased-array 3-D pencil beam radar utilizing a null-to-null cross-type scan, which scans the target as it moves. The simulation varies the location of the target in the scanning window by use of a uniform random number generator before the start of each scan. The scanning window is always wide enough to fully scan the target.

The simulation executes a single scan on the moving target and then increments time to allow the radar to perform its other search and track duties, and to move the target down the glidepath. The simulation aborts when the target is within 90 meters of the runway touchdown point.

The target model used is an ensemble of three anisotropic scattering complexes representing the left wing, right wing, and fuselage. The location of the scattering complexes in the target coordinate system is shown in Figure 2-2(a), and the arrangement of the scattering points in a scattering complex is shown in Figure 2-2(b). The equations describing the scattering complexes are given in Table 2-1. In this work, the angles

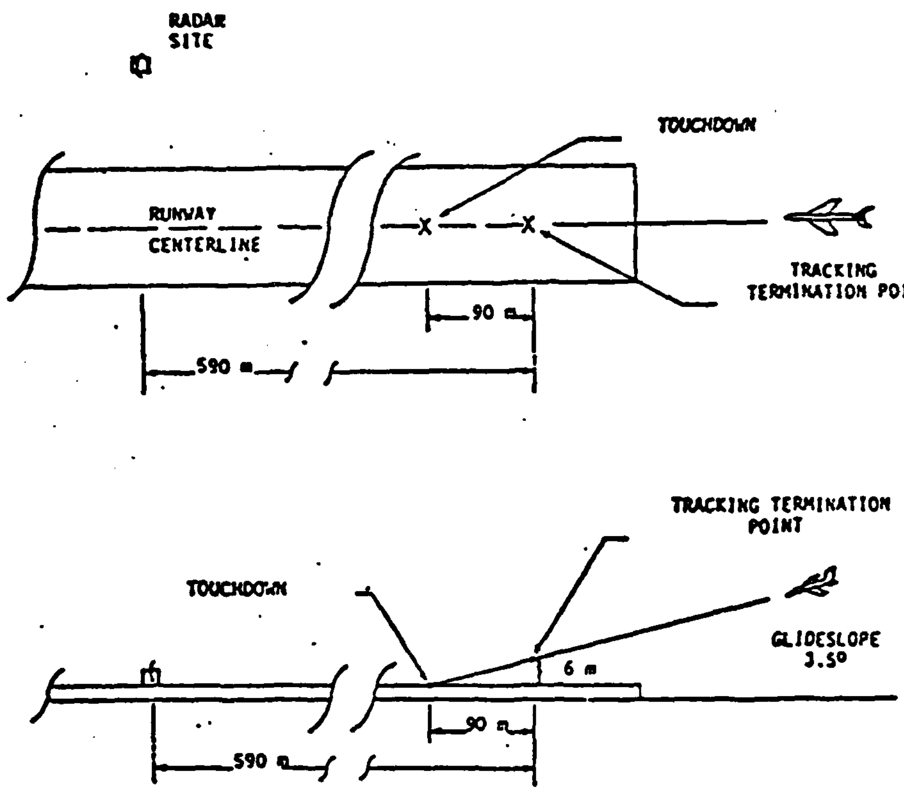
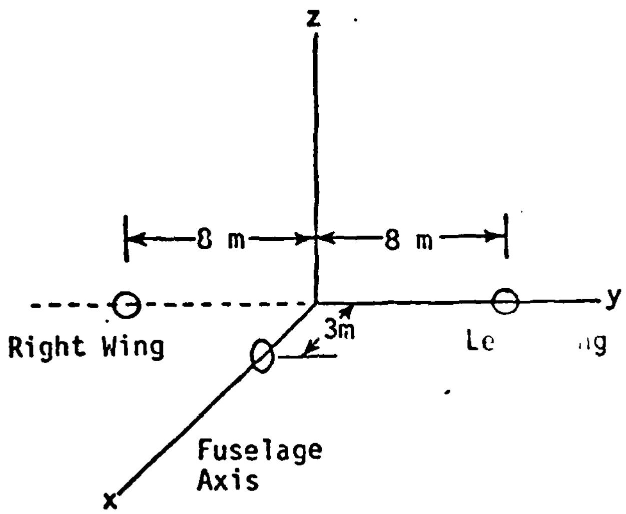
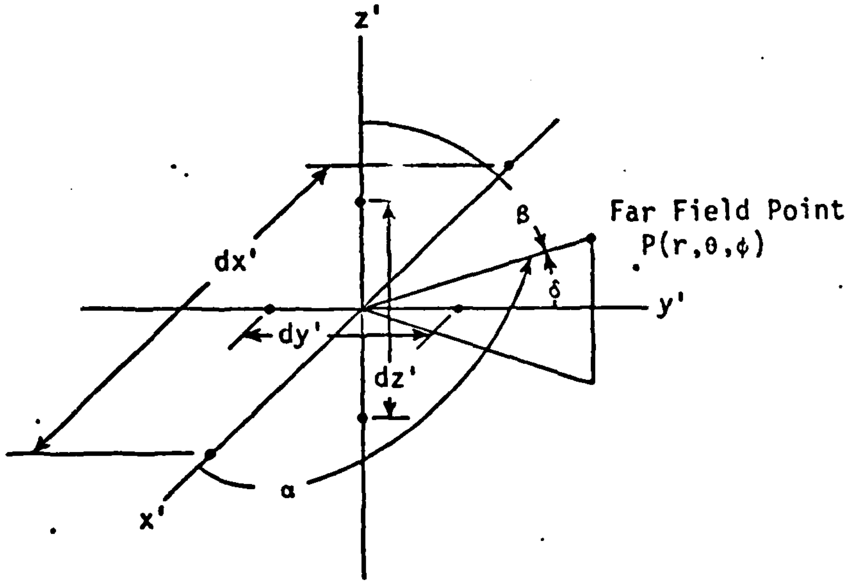


Figure 2-1. Siting of the precision approach radar and final approach glideslope used in the computer simulation.



(a)



(b)

Figure 2-2. (a) Physical orientation of the ensemble scatterers.
(b) Arrangement of the two-element scatterer complexes for the coordinate systems of the fuselage and wing scattering points.

Table 2-1. Radar cross section equations for the target model scattering complexes

RCS equation for all points:

$$\sigma(\theta,\phi) = A(\theta,\phi)\left|A_x(\alpha) + A_y(\beta) + A_z(\delta)\right| \quad (m^2)$$

where:

$$A_x(\alpha) = \cos\left(\frac{kd_x}{2}\cos\alpha\right)$$

$$A_y(\delta) = \cos\left(\frac{kd_y}{2}\cos\delta\right)$$

$$A_z(\beta) = \cos\left(\frac{kd_z}{2}\cos\beta\right)$$

α, δ, β are assumed the same for each scatterer and are defined in Figure 2-2.

Fuselage (FUS)	RH Wing (RW)	LH Wing (LW)
d_x = 10m	d_x = 6m	d_x = 6m
d_y = 2m	d_y = 4m	d_y = 4m
d_z = 2m	d_z = 2m	d_z = 2m

Amplitude Envelopes

$$A_{FUS}(\theta,\phi) = \begin{cases} (10(\theta-\pi/2)^2 + 1)\left(\frac{75}{(\pi/2)^2}\phi^2 + 8\right) & -\frac{\pi}{2} \le \phi \le \frac{\pi}{2} \\ (10(\theta-\pi/2)^2 + 1)\left(\frac{75}{(\pi/2)^2}(\pi-\phi)^2 + 8\right) & \frac{\pi}{2} < \phi < \frac{3\pi}{2} \end{cases}$$

$$A_{RW}(\theta,\phi) = (100(\theta-\pi/2)^2 + 1)(1 - \mathrm{SIN}(\phi))$$

$$A_{LW}(\theta,\phi) = (100(\theta-\pi/2)^2 + 1)(1 + \sin(\phi))$$

ϕ and θ are not the spherical phi and theta, but rather relative angles measured from the nose axis of the target coordinate system. Phi describes that angle in azimuth, and theta describes the angle in elevation. Figures 2-3, 2-4, and 2-5 are plots of the radar cross section (RCS) in azimuth of the fuselage, right wing, and left wing, respectively. The composite cross sections of the target model in azimuth, Figure 2-6, and in elevation, Figure 2-7, are not used by the simulation, and are presented here for completeness. The radar cross sections in polar form of the fuselage, right wing, and left wing, are shown in Figures 2-8, 2-9, and 2-10, respectively. The built-in shadowing effect of the fuselage on the wings is especially evident in Figures 2-9 and 2-10. The composite cross sections in azimuth, Figure 2-11, and elevation, Figure 2-12, are again shown for completeness. All figures are for a wavelength of 3.3 cm.

The individual returns from each of the scattering complexes are weighted by the antenna pattern before being summed on a power basis. This process is repeated for every simulated transmission of a pulse from the radar. Although only one pulse is transmitted at each beam pointing location, time is incremented as though three pulses are transmitted. When the simulation noise option is enabled, random gaussian noise is added to the resultant return on a power basis. The noise power is 15 dB down from a relative maximum scan (without noise) at far range.

To simulate turbulence, the target coordinate system is allowed roll, pitch, and yaw, with the origin of the target coordinate system locked on the 3.5 degree glideslope. To simulate calm air, the target model maintains a "wings level" attitude for the duration of the flight.

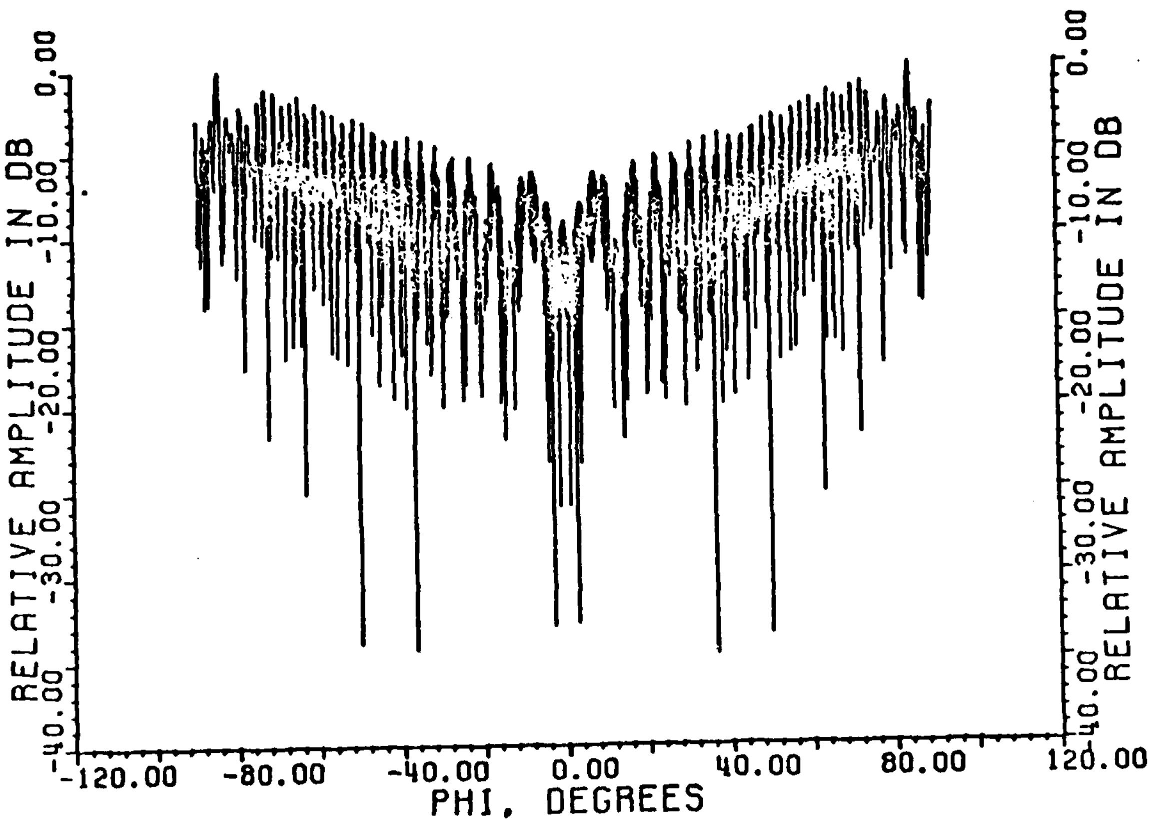


Figure 2-3. Radar cross section of fuselage scattering complex in azimuth, with the azimuth angle measured from the nose axis of the coordinate system.

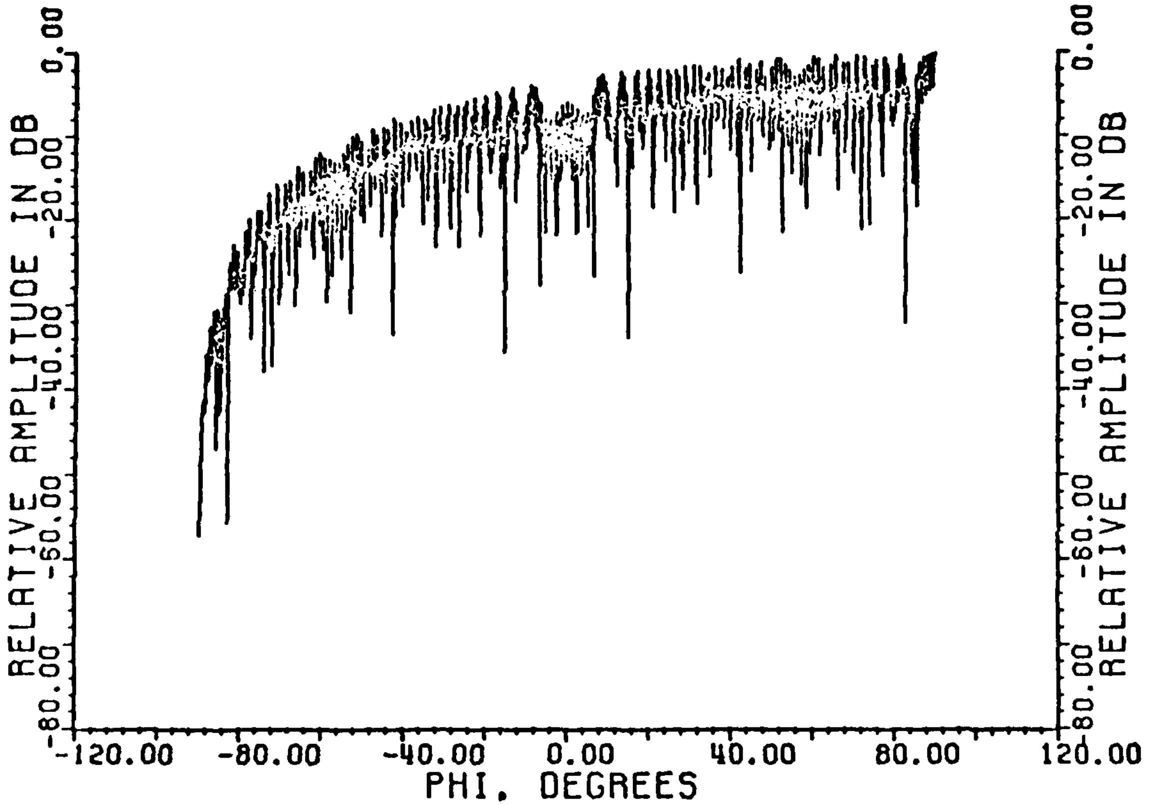


Figure 2-4. Radar cross section of *right wing scattering complex in* azimuth, with the azimuth angle measured from the nose axis of the coordinate system.

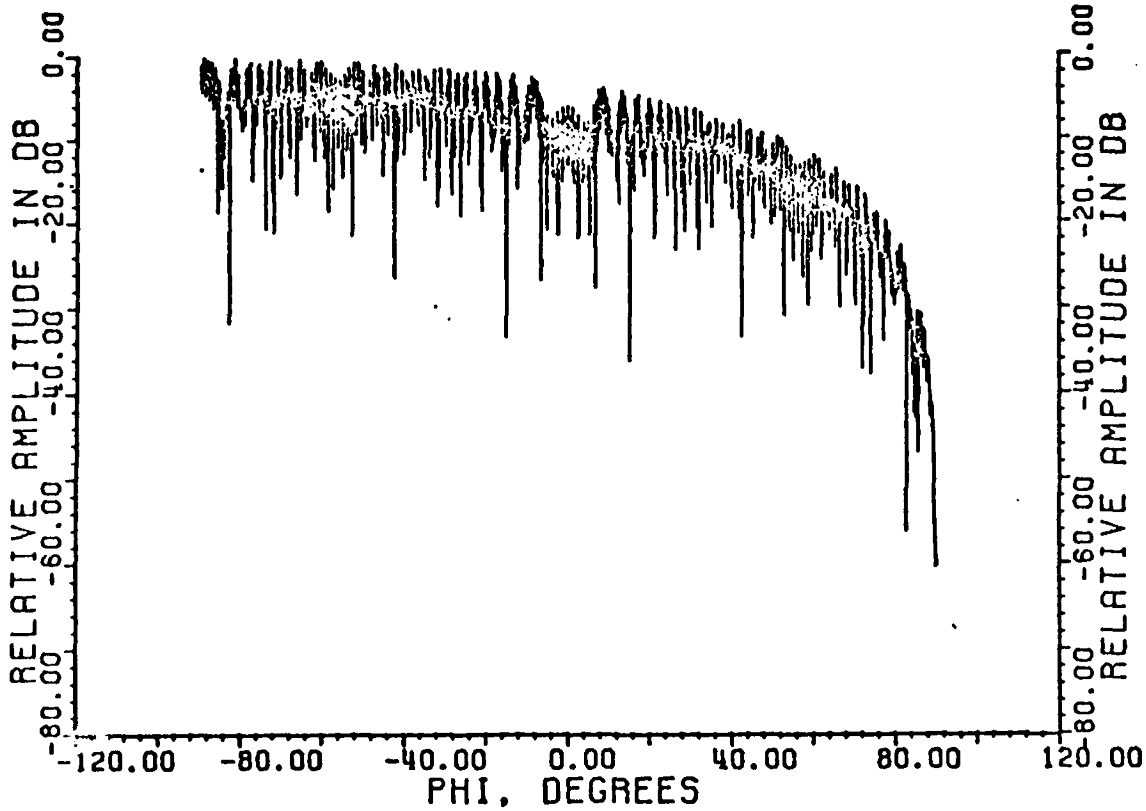


Figure 2-5. Radar cross section of left wing scattering complex in azimuth, with the azimuth angle measured from the nose axis of the coordinate system.

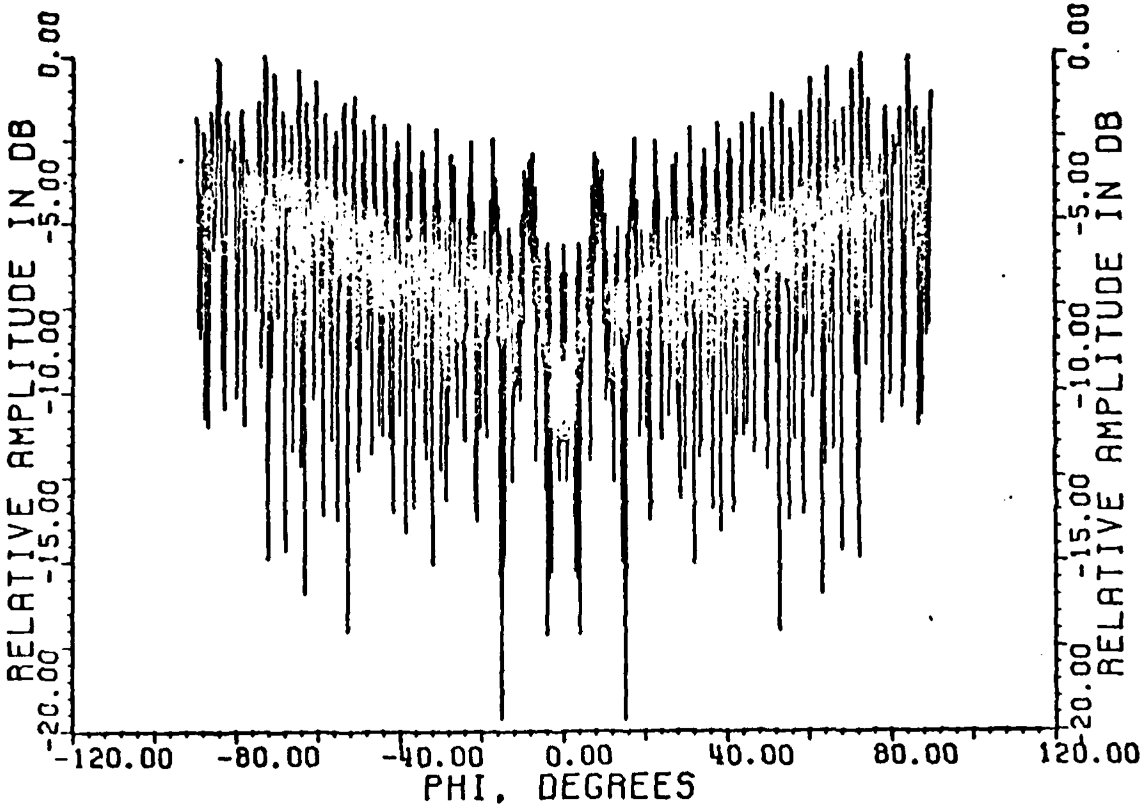


Figure 2-6. Composite cross section in azimuth, with the aximuth angle measured from the nose axis of the coordinate system.

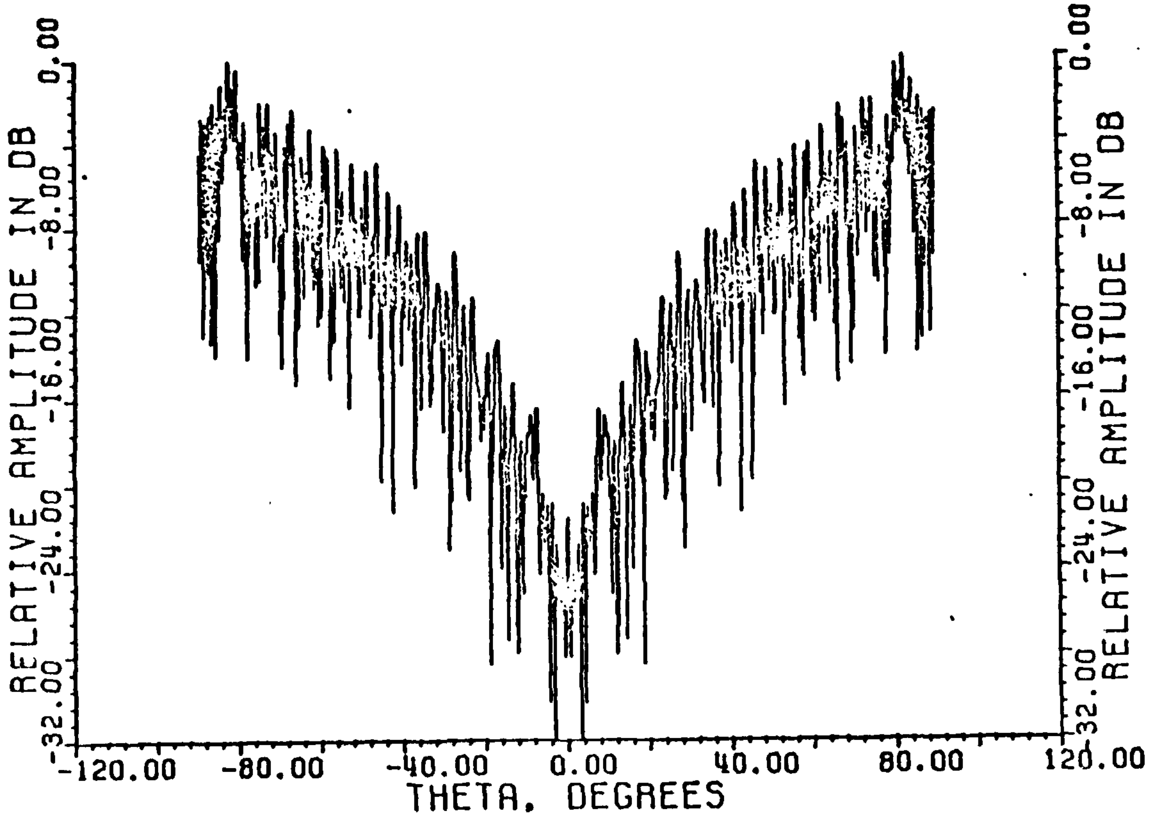


Figure 2-7. Composite cross section in evaluation, with the evaluation angle measured from the nose axis of the coordinate system.

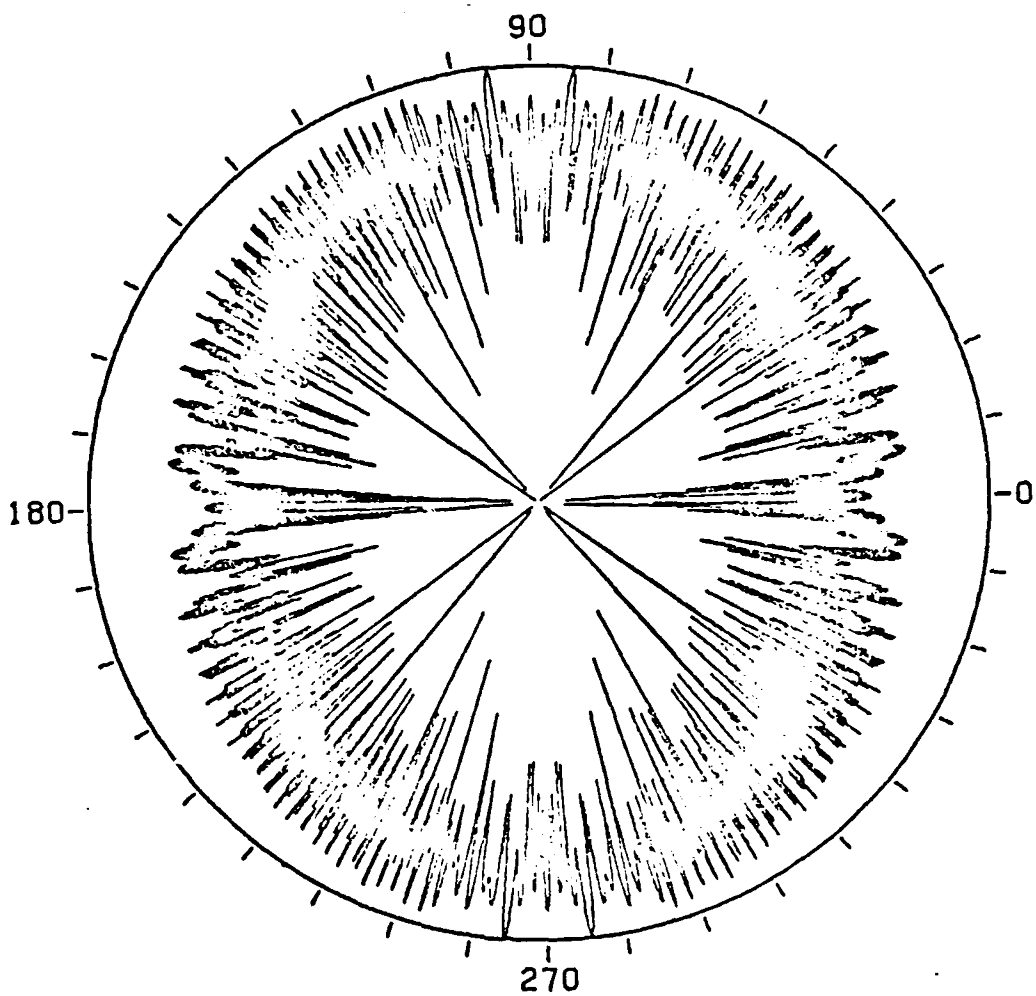


Figure 2-8. Radar cross section of fuselage scattering complex in azimuth, with the azimuth angle measured from the nose axis of the coordinate system. Amplitudes are in dB down from maximum.

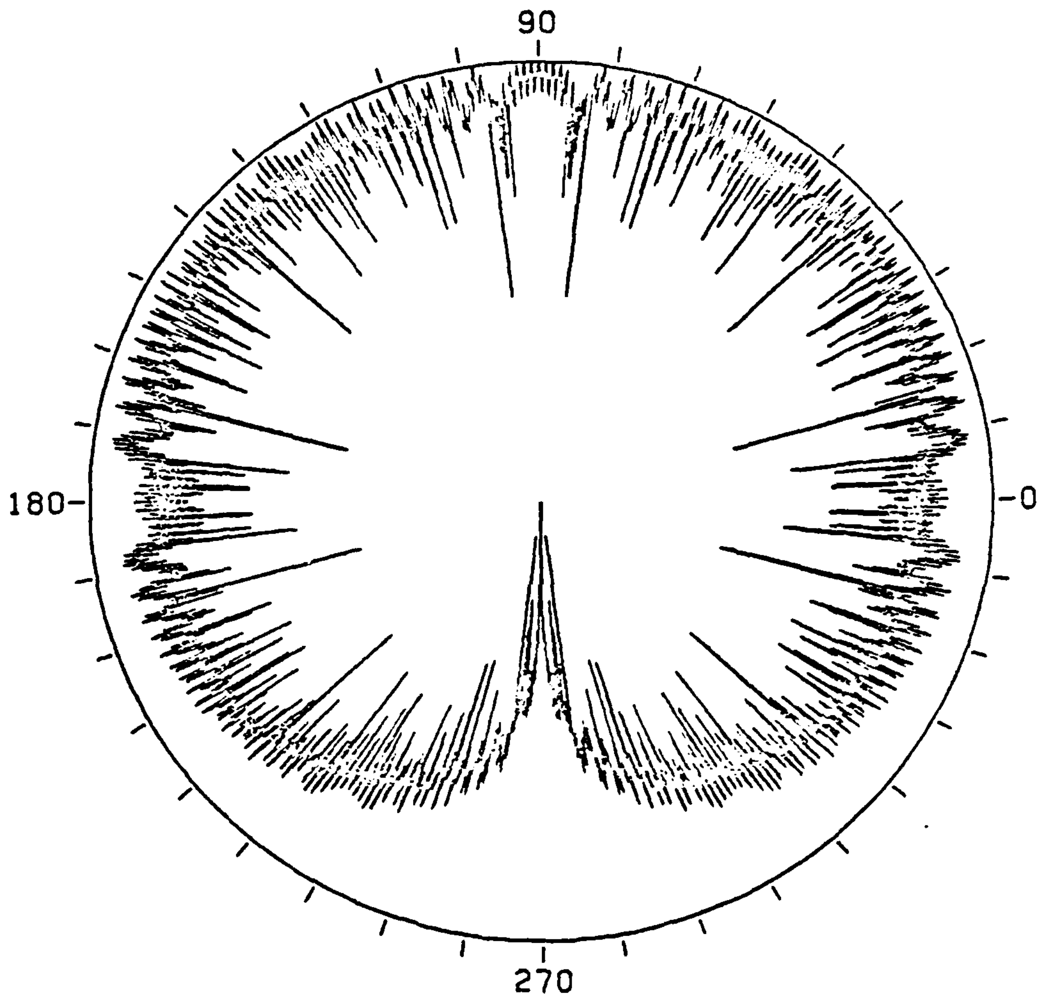


Figure 2-9. Radar cross section of right swing scattering complex in azimuth, with the azimuth angle measured from the nose axis of the coordinate system. Amplitudes are in dB down from maximum.

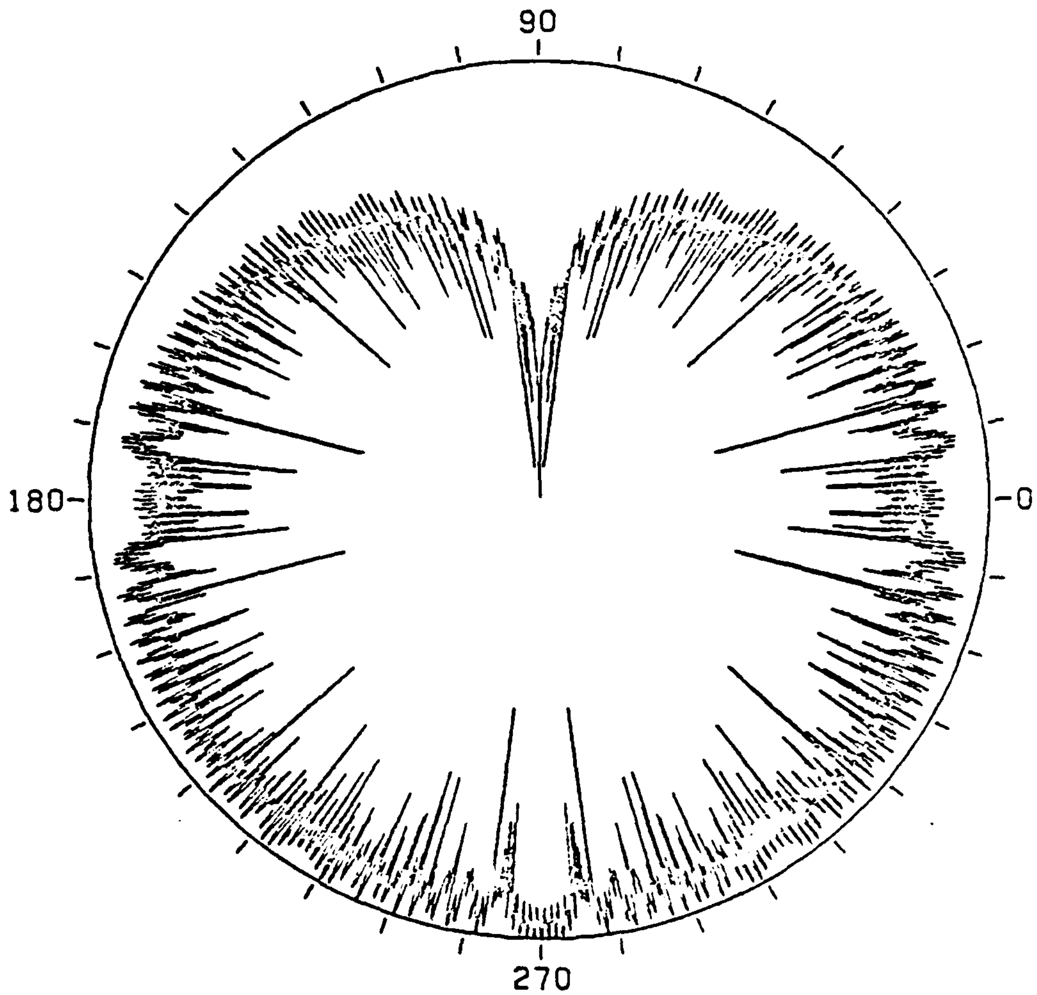


Figure 2-10. Radar cross section of left wing scattering complex in azimuth, with the azimuth angle measured from the nose axis of the coordinate. Amplitudes are in dB down from maximum.

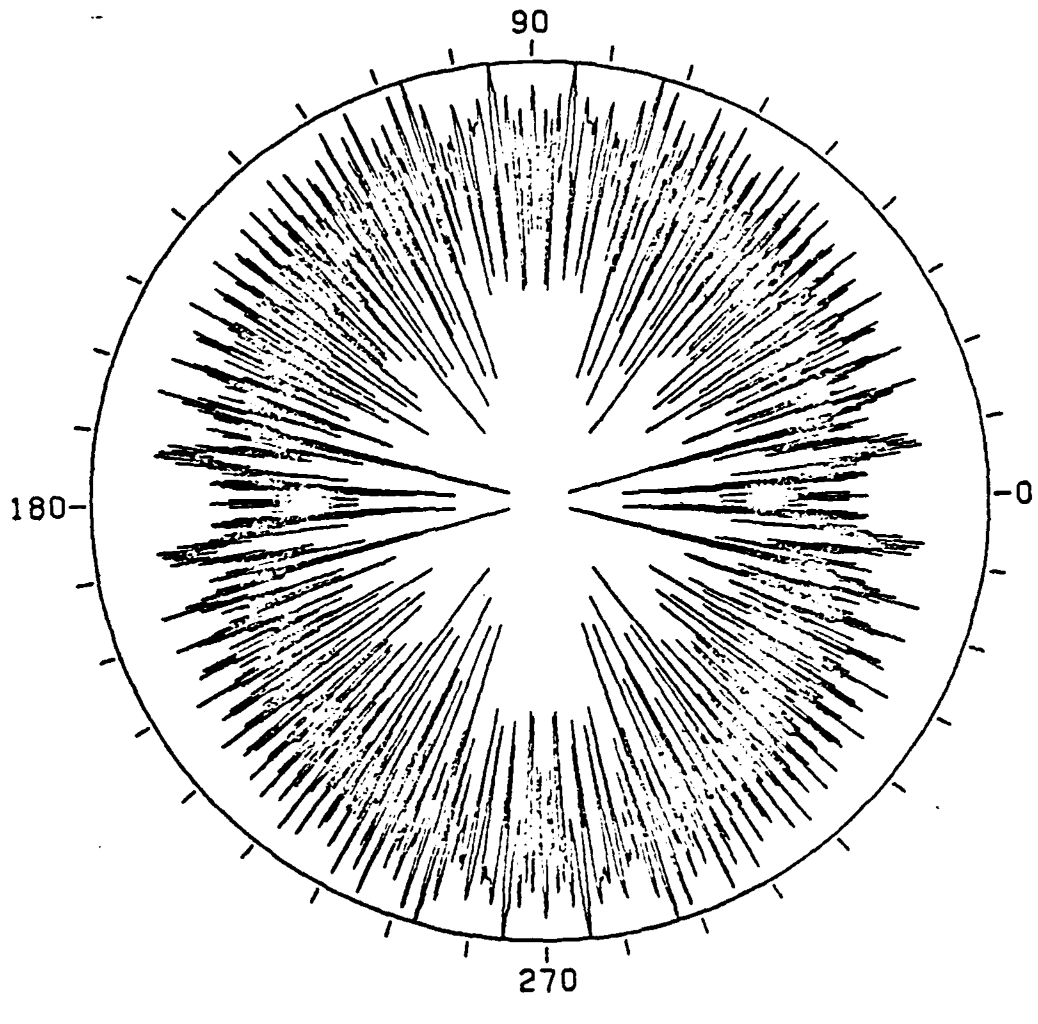


Figure 2-11. Composite cross section in azimuth, with the aximuth angle measured from the nose axis of the coordinate system. Amplitudes are in dB down from maximum.

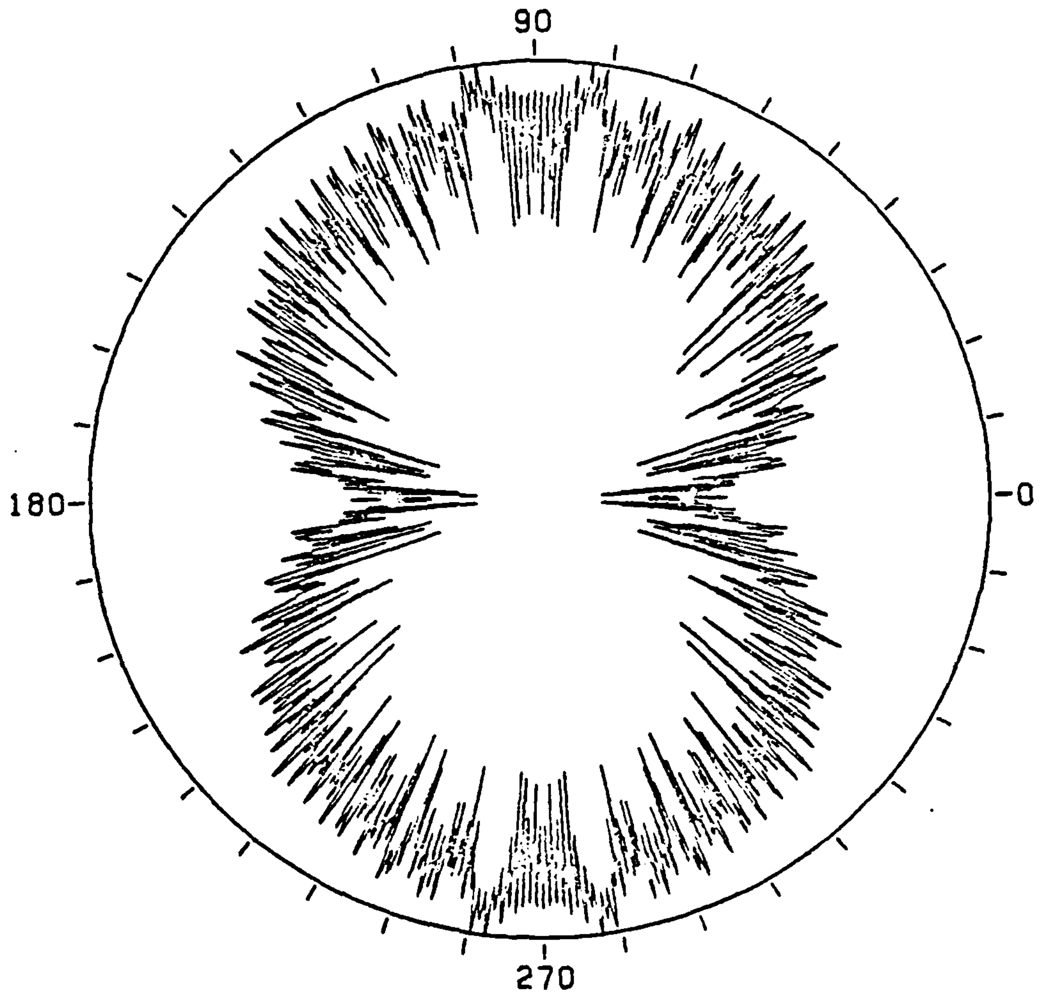


Figure 2-12. Composite cross section in evaluation, with the evaluation angle measured from the nose axis of the coordinate system. Amplitudes are in dB down from maximum.

The target returns are calculated with the simplified form of the radar equation, and are output to the centroid estimators. The basic system parameters are listed in Table 2-2.

Table 2-2. Parameters for landing system simulation.

Frequency	9.1 GHz
Pulse repetition frequency	6 KHz
Target initial elevation	56.6 mrad
Target model initial range from touchdown	6890 meters
Target model speed	148.6 mph
Turbulence rates	10 deg/s roll 5 deg/s yaw and pitch
Signal-to-noise ratio at far range	15 dB
Antenna beamwidth (null-to-null)	
Azimuth	1.83°
Elevation	1.77°
Simulation duration	103 scans

III. SIGNAL PROCESSING

The computer simulation just described creates a sequence of scan returns from the target. In order to neglect the effects of multipath, this work will address itself solely to that data generated by the scan in azimuth. The target centroid is calculated from the returns as follows. A threshold determined from the scan returns is applied to the scan. Moving in from the edges of the scan, the first occurrance of two consecutive return voltages exceeding the threshold is located. The outermost of those return voltages are tagged as the edge-points of the target. Since the angle to the returns are known, the centroid of the target is judged to be midway between the edge-points.

Three methods of setting the threshold are used in this work. Two are the mean, the median, post-determined thresholds. That is, the target is scanned and the returns are recorded. The mean of the scan returns is calculated, and a threshold is set at that level. Likewise, the median scan return is found and a threshold is set at that level.

A third method is a pre-determined thresholding method. The antenna beam is placed in the center of the scanning window to measure the anticipated maximum return from that scan. The threshold is set 12 dB down from that return level. When two consecutive returns are above the 12 dB threshold, the edge is marked and the scanning translates to the other side to determine the other edge-point. The requirement that the target be fully scanned no longer exists for this method, so that fewer pulses are needed to locate the target.

A fourth method used is a non-thresholding technique, the radar centroid (Radar CG). This estimator weights each antenna pointing angle in the scanning window by the return from that angle, and divides the sum of the weighted angles by the sum of the weights (returns). The result is the angle to the radar center of gravity of the body of the return. Since it requires that the window be fully scanned, all available pulses are used.

These four methods of centroid location have been compared to the new method, centroid location based upon return amplitude-versus-angle signature.

IV. THE NEW TARGET CENTROID ESTIMATING ALGORITHM

Introduction

Since all target centroid estimators are based on scan returns, it is instructive to examine the flight scan-return history of a target. Figure 4-1 is the scan return history of the model in still air without noise added, which shall now be referred to as a baseline flight. This plot was made with the target in the center of the scanning window. The first and last beam pointing locations have negligible return amplitudes since a null-to-null cross track is employed; the first null in the antenna pattern is placed on the target at those beam locations. As is to be expected, the maximum return occurs in the center of the scan. It is readily seen that the scan returns over the flight are modulated, specifically by the scintillation of the target model radar cross section. In particular, note scan number 90. At this scan, the antenna is clearly in a null of the target RCS. We can also pick out scans 78, and with greater difficulty, scan 58, as being in nulls of the target model cross section. It is in these scans, with poor target returns, that we would expect the target location error of the estimators to increase.

A flight with noise is shown in Figure 4-2. The two large bodies of return between scans 58 and 90 are still clearly seen, but the effect of noise is pronounced on the rest of the flight. Beam pointing locations 1 and 49 are no longer at zero amplitude, but are raised (or lowered) in level by noise. It is clearly seen from observation of scans 90, 78, and

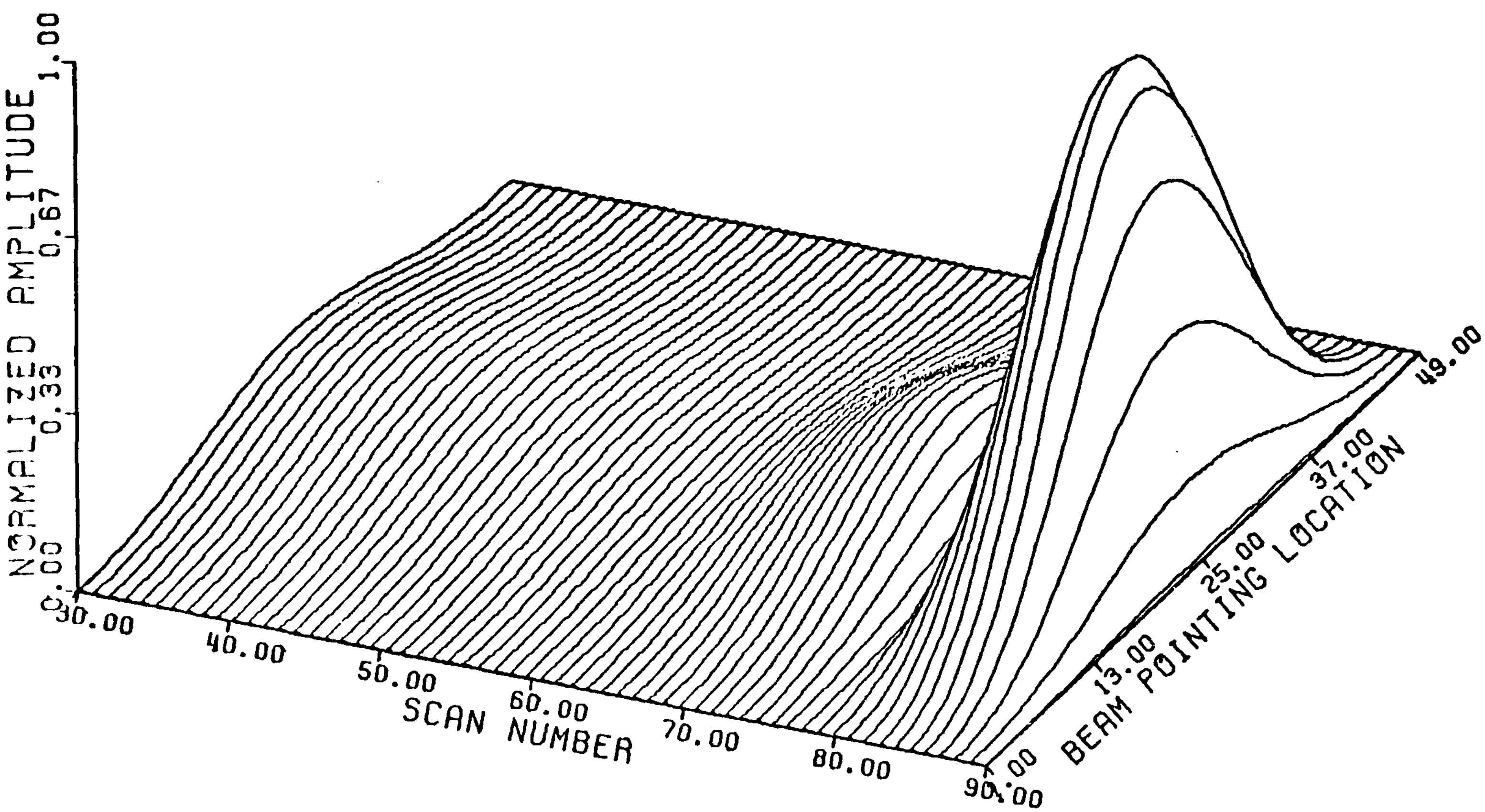


Figure 4-1. Target return history for a baseline flight with 49 beam pointing locations in the scanning window, with the target in the center of the cross track, azimuth scan. Shown is a portion of the flight from the 30th to the 90th scan, inclusive.

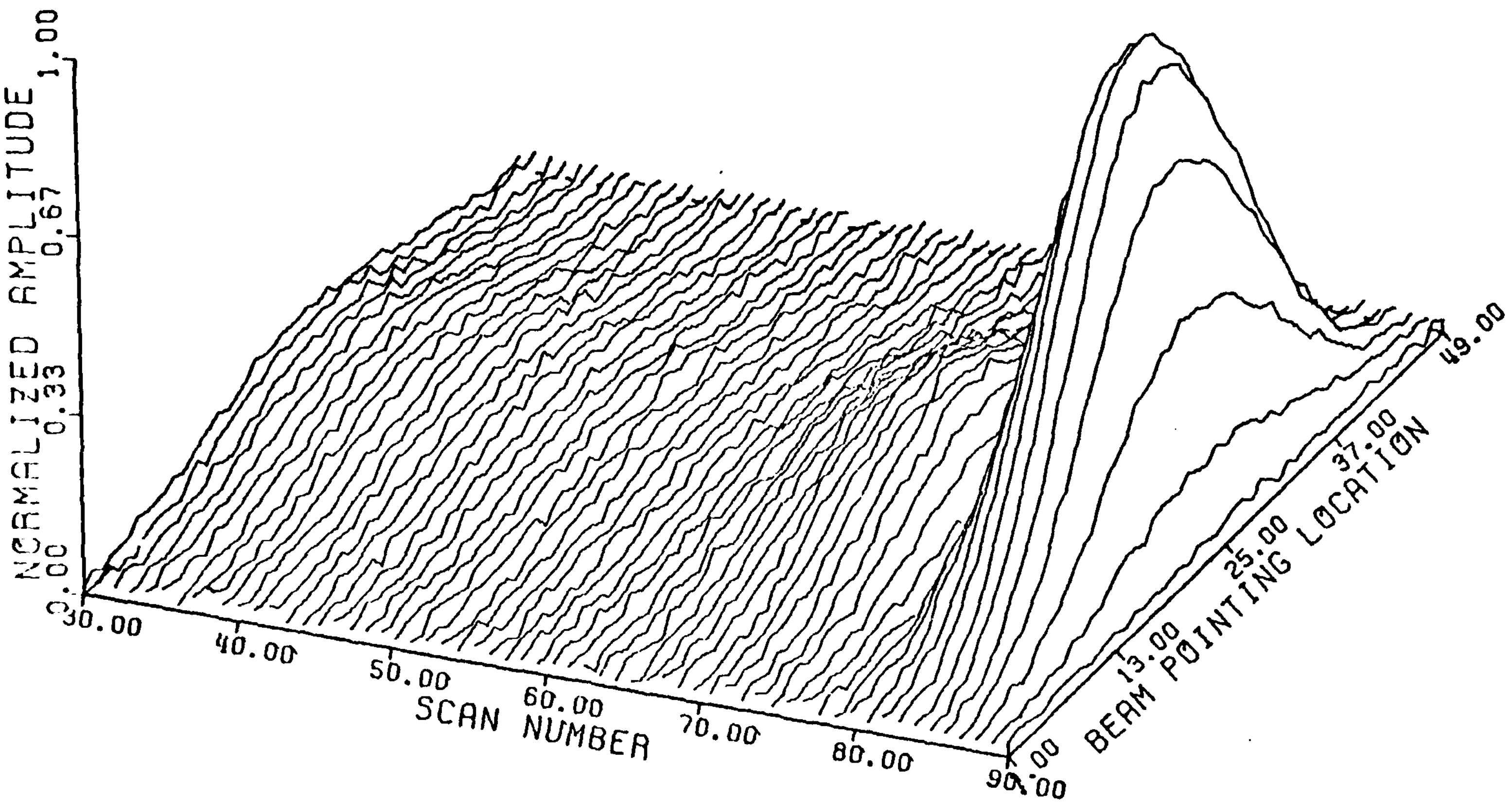


Figure A-2. Target return history for a baseline flight with noise added, 49 beam pointing locations in the scanning window, target in the center of the cross track, azimuth scan. Shown is a portion of the flight from the 30th to the 90th scan, inclusive.

58 that an accurate determination of the presence of a target at those scans would be very difficult and prone to error, whereas the detection of the target with a good signal return, even in the presence of noise, is less prone to error.

Figure 4-3 is of a baseline flight with turbulence. The many nulls in this plot are the result of the modulation of the target model radar cross section on the target returns as model rotates on its axis in simulation of turbulent wind conditions. Again, beam locations 1 and 49 exhibit negligible returns as the null in the antenna pattern is on the target.

Addition of noise to the flight with turbulences is shown in Figure 4-4. The many returns that were of low signal level are now filled in with noise. Only those scans whose signal level rises above the noise are suitable for target detection.

It is in this light that the work towards a new centroid algorithm was conducted. The algorithm must be able to determine which scans are suitable for target detection and location - and to discard all others.

The Algorithm

It is observed in Figure 4-1, which graphically depicts the scan history of a baseline flight, that all scan envelopes have a high degree of symmetry. That is, as the antenna beam illuminates the target first with the pattern null, then increasing the illumination as the main lobe moves onto the target, reaching the maximum when the beam is centered on the target, then dimenishing as the target is placed in the pattern null, the overall scan envelope takes on a bell shape due to the modulation of the antenna beam. Since the return envelopes are of this shape, each side

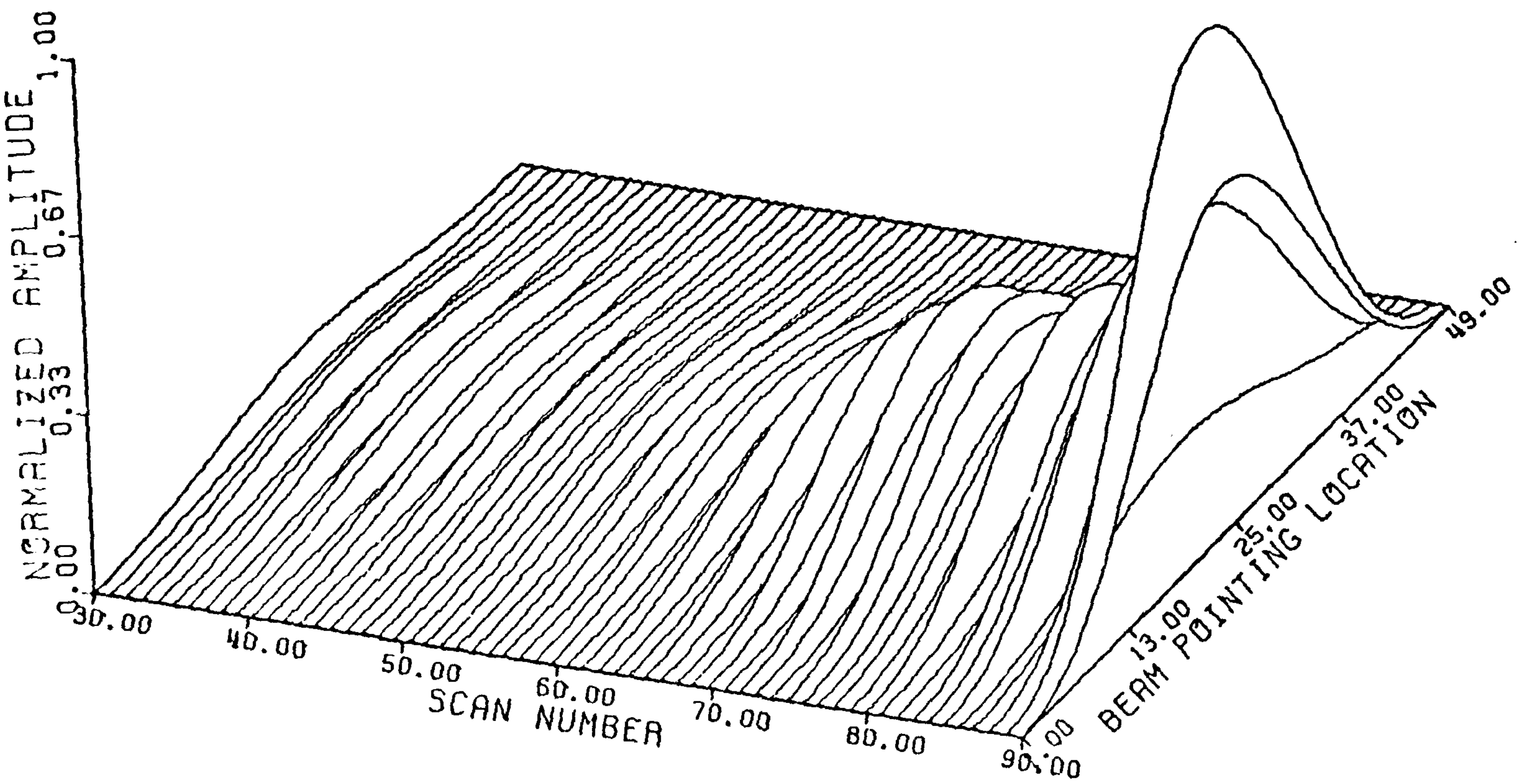


Figure 4-3. Target return history for a baseline flight with turbulence added, 49 beam pointing locations in the scanning window, target in the center of the cross track, azimuth scan. Shown is a portion of the flight from the 30th to the 90th scan, inclusive.

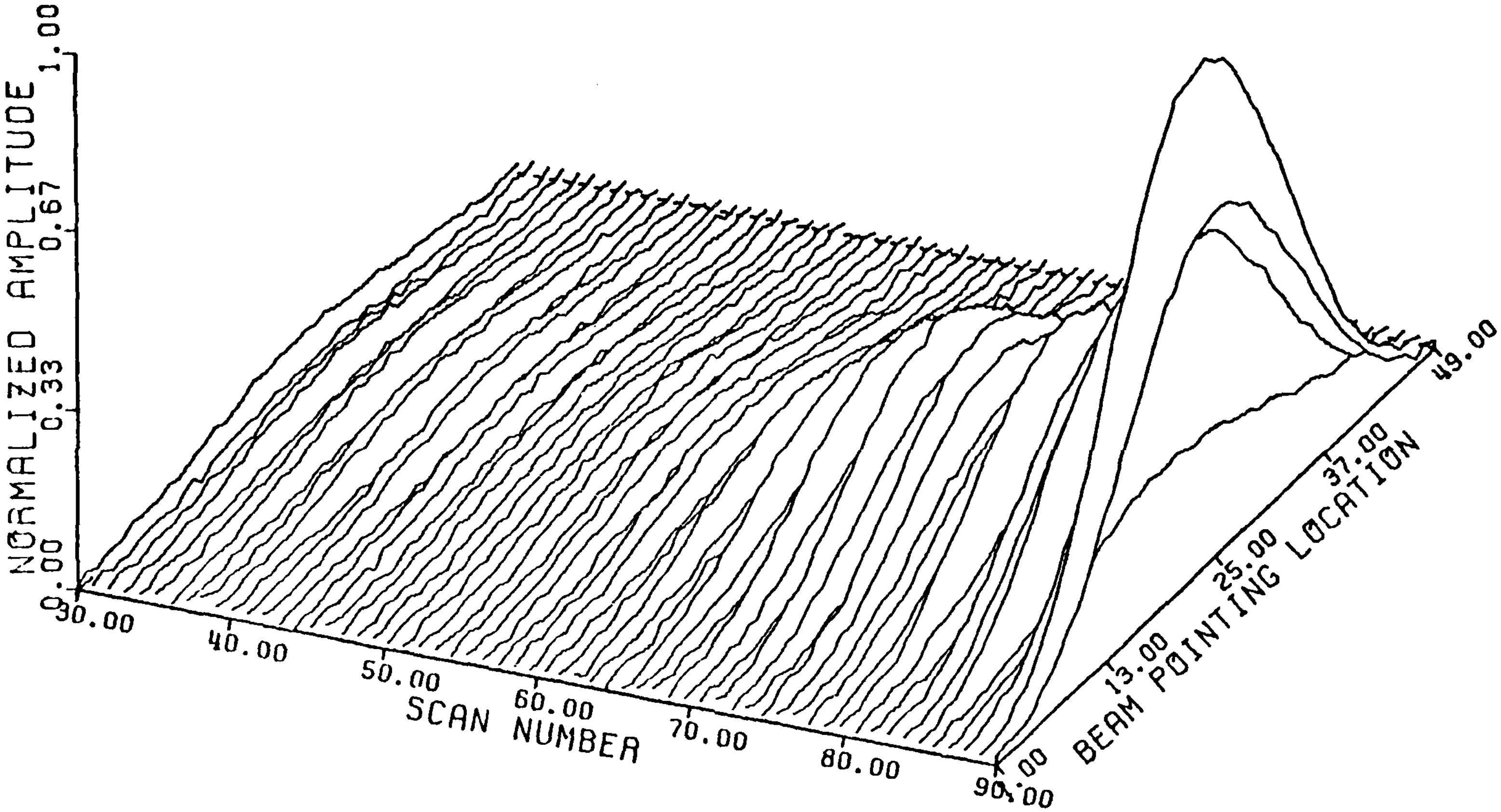


Figure 4-4. Target return history for a baseline flight with noise and turbulence added, 49 beam pointing locations in the scanning window, target in the center of the cross track, azimuth scan. Shown is a portion of the flight from the 30th to the 90th scan, inclusive.

of the bell shape has a unique point, the point of maximum slope. Returning to Figure 4-2 it is observed that the maximum slope of a scan with a low SNR (such as scans 58, 78, and 90) is relatively small, and those scans with large SNRs have a relatively large maximum slope. This, then, is the chosen criterion: Find the point of maximum slope; compare the slope at that point to a minimum acceptable value and set the edges of the target or centroid.

The method used to find the point of maximum slope is based on the scan shape. Referring to Figure 4-5, let us assume that we are using a cross track with a granularity of 5 beam pointing locations. The relative amplitudes of the extended returns are marked by the lettered X's on the drawing. Moving from left to right, the first three returns have a positive second derivative, since the slope $\overline{BC}$ is greater than slope $\overline{AB}$. Points B, C, and D have a negative second derivative, since slope $\overline{CD}$ is less than slope $\overline{BC}$. Since the point of maximum slope is where the second derivative is zero, that is, where the second derivative changes sign, the maximum slope must have occurred between points B and C. Having found the maximum slope, we check to ensure that its' magnitude is greater than the minimum acceptable slope. If it is, the target edge is marked as being midway between points B and C, and scanning translates to the other side of the scan. The process is then repeated for returns G, F, E, and D. When the two target edges are found, the centroid is placed midway between the edge points. Since the target is located by calculating second derivatives, this method is referred to in this work as the second derivative method or SDRV.

A thorough comparitive analysis of this method will be presented in a forthcoming report.

TARGET CENTROID LOCATION

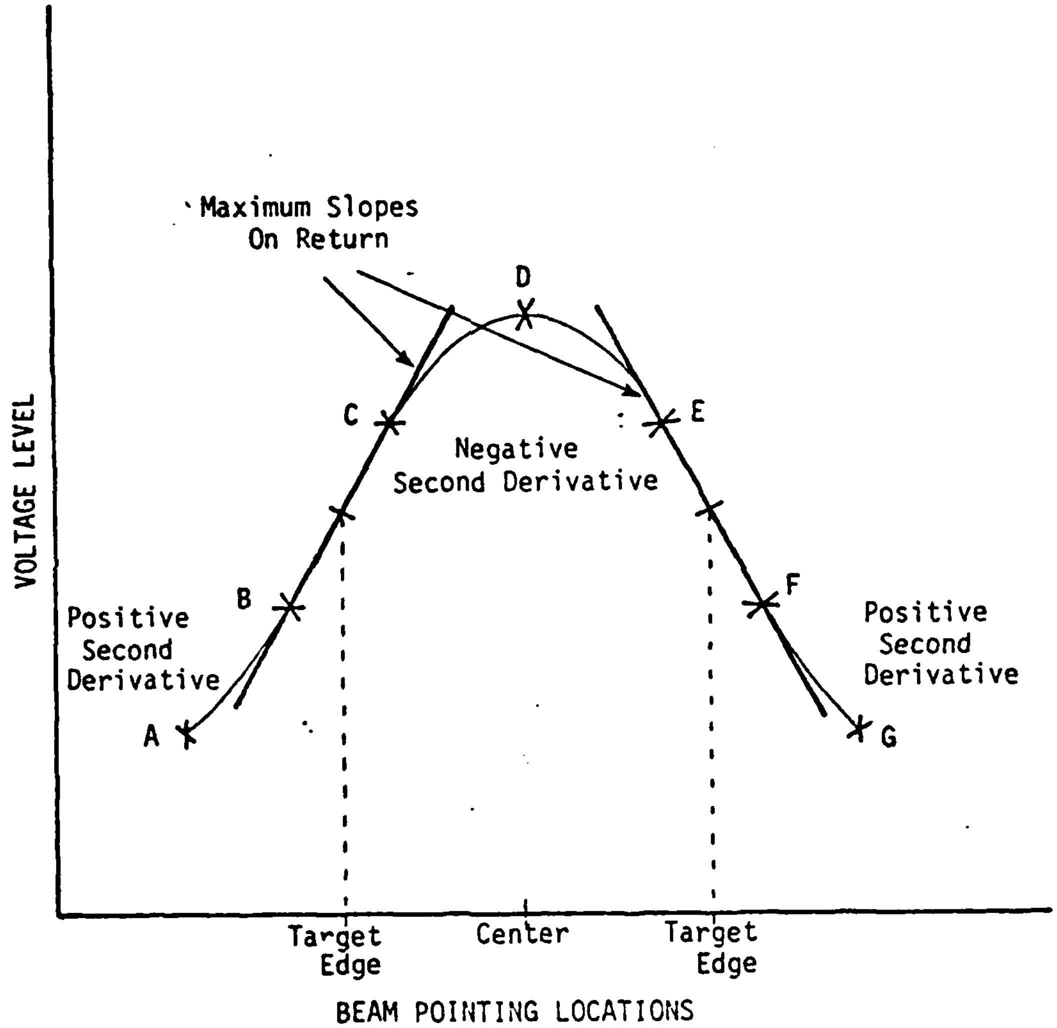


Figure 4-5. Illustration of the method employed to determine the target centroid location based on the shape of the scan envelope. The signal returns are marked by X.

V. Conclusion

The fundamental features of the new centroid algorithm were presented against the background of the cross-scan tracking technique. Analysis presently being concluded seems to indicate that the new method is generally both more accurate and robust than the techniques used for comparison.

PART FOUR

ADAPTIVE FILTERING ALGORITHMS FOR THE MATCALS SYSTEM

Prepared for

Georgia Institute of Technology
ATLANTA, GEORGIA

Under

Contract 1-A-2550

by

Electrical Engineering Department
Auburn University
Auburn, Alabama

Prepared by: G. Mack Lee and Mark C. Tatum

Reviewed by: Scott A. Starks

I. INTRODUCTION

When using an AN/TPN-22 track-while-scan radar to track an approaching aircraft, raw digitized measurements of position are obtained. This positional information is filtered and processed by a control system to determine command signals which are used by the controller to correct or compensate for deviations of the aircraft's position from the prescribed glideslope path.

In such a system, the positional information produced by the radar must be processed to yield a smoothed present-position estimate, a smoothed present-position estimate, a smoothed present velocity estimate, and a one step ahead predicted position for track correlation or bin selection. Filters which accomplish these goals are referred to as tracking filters.

In track-while-scan systems, the tracking filters are realized in the form of digital filters with fixed or time-varying coefficients. Due to the presence of noise in such a system, it becomes necessary to design the filters based upon the criteria of good noise smoothing and good maneuver following. Typically systems with good noise smoothing characteristics have sluggish system response which prohibits them from following targets with rapidly changing dynamics. In contrast, systems possessing good maneuver-following characteristics have large bandwidths and thus poor noise smoothing ability. Thus, in designing a tracking filter a compromise must be made between these two conflicting goals. One may

do so by designing an adaptive filter whose noise-smoothing and maneuver-following characteristics adapt to the current dynamics of the aircraft. Two approaches are presented. The first approach is based upon adaptively selecting the output from either a fixed parameter α-β on a fixed parameter α-β-γ filter. This selection is determined by an algorithm which incorporates an estimate of the tracking error correlation coefficient. The second approach is based upon an algorithm which automatically adjusts the parameters of an α-β filter to adapt to the dynamics under track.

II. BI-STATE ADAPTIVE FILTER

When tracking a moving target such as an aircraft, the positional data obtained from the radar return is corrupted by noise. These noisy positional measurements can be digitally filtered to provide smoothed estimates of position and velocity. One type of filter which may be used to smooth the noisy data is a fixed parameter α-β filter. The fixed parameter α-β tracking filter employs an approach that, if handled properly, will provide good results with a minimum amount of computation. The α-β filter is a narrow bandwidth filter which provides good noise smoothing capabilities based on the assumption that the aircraft flies a constant velocity, straight-line trajectory. The prediction equation is a simple linear extrapolation and the expressions for smoothed position and velocity use simple gain terms α and β to weigh the effects of differences between the measured and predicted positions. The α-β filter equations will merely be stated in this paper and the interested reader is directed to the referenced literature [1].

$$x_s(k+1) = x_p(k+1) + \alpha[x_m(k+1) - x_p(k+1)] \tag{4-1}$$

$$\dot{x}_s(k+1) = \dot{x}_p(k+1) + (\beta/T)[x_m(k+1) - x_p(k+1)] \tag{4-2}$$

$$x_p(k+1) = x_s(k) + T\,\dot{x}_s(k) \tag{4-3}$$

$$\dot{x}_p(k+1) = \dot{x}_s(k) \tag{4-4}$$

where

$x_s(k+1)$ = smoothed position at time k+1

$\dot{x}_s(k+1)$ = smoothed velocity at time k+1

$x_p(k+1)$ = predicted position at time k+1

$\dot{x}_p(k+1)$ = predicted velocity at time k+1

$x_m(k)$ = measured position at time k

T = sampling interval between times k and time k+1

It is shown in [2] that for a fixed parameter α-β filter, α and β are found via (4-5) - (4-7).

$$K = \frac{L-1}{L+1} \tag{4-5}$$

where $L \triangleq$ effective length of filter window.

$$\beta = 1 - K^2 \tag{4-6}$$

$$\alpha = (1-K)^2 \tag{4-7}$$

Unfortunately, if the aircraft deviates from its straight-line constant velocity trajectory the fixed parameter α-β filter will be in error.

Another approach to tracking a moving aircraft is to use a filter based on a more generalized model of the aircraft's trajectory. A more generalized model of a maneuvering aircraft's trajectory may be obtained by incorporating third order prediction equations in the tracking filter. One such third order tracking filter is the fixed parameter α-β-γ filter. While α and β in the α-β-γ tracking filter perform the same function as they did in the fixed parameter α-β filter, the γ term brings into play the much needed acceleration estimate essential to tracking the maneuvering aircraft. This more generalized tracker not only maintains track

throughout a maneuver or turbulent condition, but in addition, provides good estimates of position and velocity with very little increase in computational difficulty. The equations for the α-β-γ filter are merely stated in this paper. The interested reader is referred to [2] for a more detailed study.

$$x_p(k) = x_s(k-1) + T\,\dot{x}_s(k-1) \tag{4-8}$$

$$\dot{x}_p(k) = \dot{x}_s(k-1) + T\,\ddot{x}_s(k-1) \tag{4-9}$$

$$E(k) = x_m(k) - x_p(k) \tag{4-10}$$

$$x_s(k) = x_p(k) + \alpha E(k) \tag{4-11}$$

$$\dot{x}_s(k) = \dot{x}_p(k) + (\beta/T)\,E(k) \tag{4-12}$$

$$\ddot{x}_s(k) = \ddot{x}_s(k-1) + (\gamma/T^2)\,E(k) \tag{4-13}$$

where,

$x_p(k)$ = predicted position estimate

$\dot{x}_p(k)$ = predicted velocity estimate

$E(k)$ = error between predicted and measured position

$x_s(k)$ = smoothed position estimate

$\dot{x}_s(k)$ = smoothed velocity estimate

$\ddot{x}_s(k)$ = smoothed acceleration estimate

The short time constant and high bandwidth of the α-β-γ filter insures trackability through a maneuver. However, if the aircraft is not maneuvering, these same characteristics contribute a significant

degradation in performance compared to the simpler α-β tracker that anticipates the constant velocity straight-line motion.

It must not be apparent that to accurately track a target in motion will require the use of the α-β filter (for straight-line constant velocity trajectories) and the α-β-γ filter (for maneuvers or turbulence). Therefore, it is clear that some method of adaptively selecting the appropriate filter output for the maneuvering/non-maneuvering cases must be found.

One method of intelligently selecting the appropriate outputs of the α-β or α-β-γ trackers is the evaluation of the tracking error correlation coefficient. A system diagram depicting this approach is shown in Figure 4-1. The α-β and α-β-γ filters operate in parallel with one another. Prediction error estimates are generated from both filters by differencing the present predicted position and the present measured position. As the prediction error estimates are generated they are stored in an L-length shift register (L being the effective window length of the filters.) From these prediction error estimates the tracking error correlation coefficient for each filter is calculated. If the tracking error is due solely to the radar quantization noise, which is white zero-mean Gaussian noise, then this error should be uncorrelated. If on the other hand the tracking error is due to positional error (maneuver) then the tracking error will be highly correlated. The error correlation coefficients are compared to a predetermined threshold and the appropriate filter output is selected based on the following premises:

1) If the correlation of error is low ($\rho \approx 0$) then the error is attributed to radar quantization noise and the output of the α-β filter should be selected.

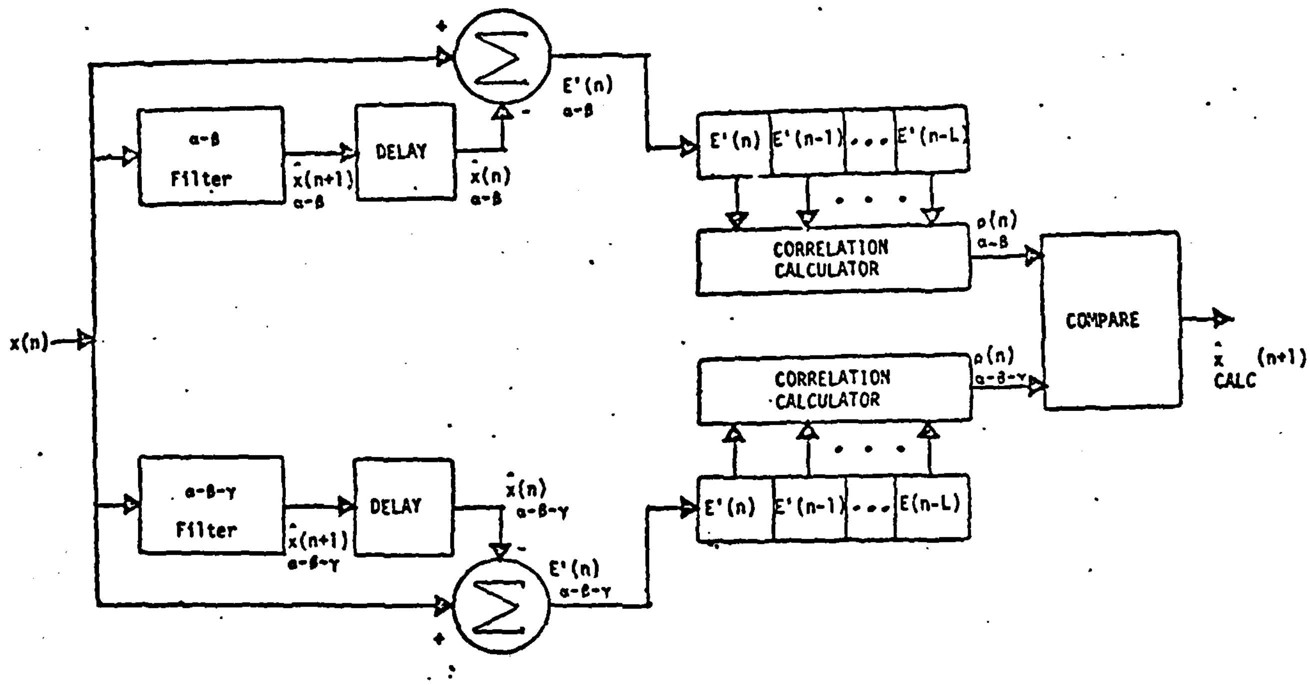


Figure 4-1. A Bi-State Adaptive Filter

2) If the correlation of error is high ($\rho > V_T$) then the error is due to positional error (maneuver) rather than radar quantization noise and the output of the α-β-γ filter should be selected.

The equations for the prediction errors and the correlation coefficient are given below.

$$E'_{\alpha-\beta}(n) = \hat{x}_{\alpha-\beta}(n) - x(n) \tag{4-14}$$

$$E'_{\alpha-\beta-\gamma}(n) = \hat{x}_{\alpha-\beta-\gamma}(n) - x(n) \tag{4-15}$$

$$\rho = \frac{\frac{1}{L-1} \sum_{i=0}^{L-2} E'(n-i)\, E'(n-i-1)}{\frac{1}{L} \sum_{i=0}^{L-1} E'(n-i)^2} \tag{4-16}$$

where

$\hat{x}_{\alpha-\beta}(n)$ = output of the α-β tracker

$\hat{x}_{\alpha-\beta-\gamma}(n)$ = output of the α-β-γ tracker

$x(n)$ = input measurement

III. A RECURSIVELY ADAPTIVE α-β FILTER

A simple α-β filter with fixed parameters as described in the preceding section is severly limited when tracking a target that is undergoing a change in velocity. Such a velocity change may be the result of an intentional maneuver or merely positional changes due to turbulence or wind gusts. The inability to track during a velocity change is seen in the predictor equations which require an extremely small velocity change between sampling intervals to be accurate. From a frequency response point of view, good noise smoothing qualities for a non-maneuvering target would require the filter to have a low pass effect, (i.e., the smoothed output would depend almost entirely on the predicted values where α and β approach zero). When a change in velocity is encountered, the filter is required to increase its bandwidth and depend more on the measured values where α and β approach one, due to the errors in the predictor equations during a velocity change. A method is presented by Schooler [4] to calculate optimal values of α and β recursively. This method can be modified in order to constantly update α and β in steady state and therefore adjust the frequency response of the filter at each sampling interval to match the target's motion. Since the predictor is in error when the target accelerates, the criterion for calculating α and β is the minimization of the expected mean square error in predicted position $\overline{\varepsilon^2[x_p(k)]}$ where $\varepsilon[\cdot]$ denotes the error associated with the term in brackets and the bar denotes expected value. Realizing the predictor

equations (4-8) and (4-9) are in error during a velocity change, the error in the predicted values can be written as:

$$\varepsilon[x_p(k+1)] = \varepsilon[x_s(k)] + T\,\varepsilon[x_s(k)] + \Delta_p(k) \tag{4-17}$$

$$\varepsilon[\dot{x}_p(k+1)] = \varepsilon[\dot{x}_s(k)] + \Delta\,v(k) \tag{4-18}$$

where Δp and Δv represent the error associated with a velocity change and are assumed to have variances $\sigma_p{}^2$ and $\sigma_v{}^2$ with a covariance μ_{pv}.

Since the predicted values are linear combinations of the smoothed values, minimizing the mean square error in predicted position is equivalent to minimizing the mean square error in smoothed position and velocity. By manipulating equations (4-17) and (4-18), squaring and taking the expected value, the errors is the smoothing equations can be written as [4]:

$$\begin{aligned}\overline{\varepsilon^2[x_s(k+1)]} &= \alpha^2\sigma_m{}^2(k+1) + \{1-\alpha\}^2\ \{\overline{\varepsilon^2[x_s(k)]} + 2T\ \overline{\varepsilon[x_s(k)\dot{x}_s(k)]} \\ &\quad + T^2\ \overline{\varepsilon^2[\dot{x}_s(k)]} + \sigma_p{}^2(k)\}\end{aligned} \tag{4-19}$$

$$\begin{aligned}\overline{\varepsilon^2[\dot{x}_s(k+1)]} &= \{\beta^2/T^2\}\ \{\sigma_m{}^2(k+1) + \overline{\varepsilon^2[x_s(k)]}\} \\ &\quad - \{2\beta[1-\beta]/T\}\ \{\overline{\varepsilon[x_s(k)\dot{x}_s(k)]}\} \\ &\quad + \{1-\beta\}^2\ \{\varepsilon^2[\dot{x}_s(k)]\} - \{2\beta/T\}\ \{\mu_{pv}(k)\} \\ &\quad + \sigma_v{}^2(k) + \{\beta^2/T^2\}\ \{\sigma_p{}^2(k)\}\end{aligned} \tag{4-20}$$

where $\sigma_m{}^2(k+1)$ is the variance in measurement error at time k+1 and

[illegible] written as:

$$\overline{\varepsilon[x_s(k+1)\dot{x}_s(k+1)]} = \{\alpha\beta/T\}\ \{\sigma_m{}^2(k+1)\}$$

$$- \{1-\alpha\}\ \{\beta/T\}\ \{\overline{\varepsilon^2[x_s(k)]}\}$$

$$+ \{1-\alpha\}\ \{1-2\beta\}\ \{\overline{\varepsilon[x_s(k)\dot{x}_s(k)}\}$$

$$+ T\{1-\alpha\}\ \{1-\beta\}\ \{\overline{\varepsilon^2[\dot{x}_s(k)]}\}$$

$$- \{1-\alpha\}\ \{\beta/T\}\ \{\sigma_p{}^2(k)\} + \{1-\alpha\}\ \{\mu_{pv}(k)\}$$

Taking the partial derivatives of (4-19) and (4-20) with respect to α and β, setting equal to zero and then solving for α and β, the best values for α and β at time k+1 are [4]:

$$\alpha(k+1) = \{\overline{\varepsilon^2[x_s(k)]} + 2T\ \overline{\varepsilon[x_s(k)\dot{x}_s(k)]} + T^2\ \overline{\varepsilon^2[x_s(k)]} + \sigma_p{}^2(k)\}/$$
$$\{\sigma_m{}^2(k+1) + \overline{\varepsilon^2[x_s(k)]} + 2T\ \overline{\varepsilon[x_s(k)\dot{x}_s(k)]} + T^2\ \overline{\varepsilon^2[\dot{x}_s(k)]}$$
$$+ \sigma_p{}^2(k)\} \qquad (4\text{-}21)$$

$$\beta(k+1) = \{T\ \overline{\varepsilon[x_s(k)\dot{x}_s(k)]} + T^2\ \overline{\varepsilon^2[\dot{x}_s(k)]} + T\ \mu_{pv}(k)\}/\{\sigma_m{}^2(k+1)$$
$$+ \overline{\varepsilon^2[x_s(k)]} + 2T\ \overline{\varepsilon[\dot{x}_s(k)x_s(k)]}$$
$$+ T^2\ \overline{\varepsilon^2[\dot{x}_s(k)]} + \sigma_p{}^2(k)\} \qquad (4\text{-}22)$$

These equations provide recursive evaluation of α and β at each sampling to keep the filter adjusted to the target's type of motion. There are many ways to initilize the filter according to the amount of information known concerning the target's state at the time tracking begins. Two good methods for initilization when the targets initial velocity, position, and

acceleration are known and unknown are presented by Schooler [4]. A block diagram of how this filter can be implemented is shown in Figure 4-2.

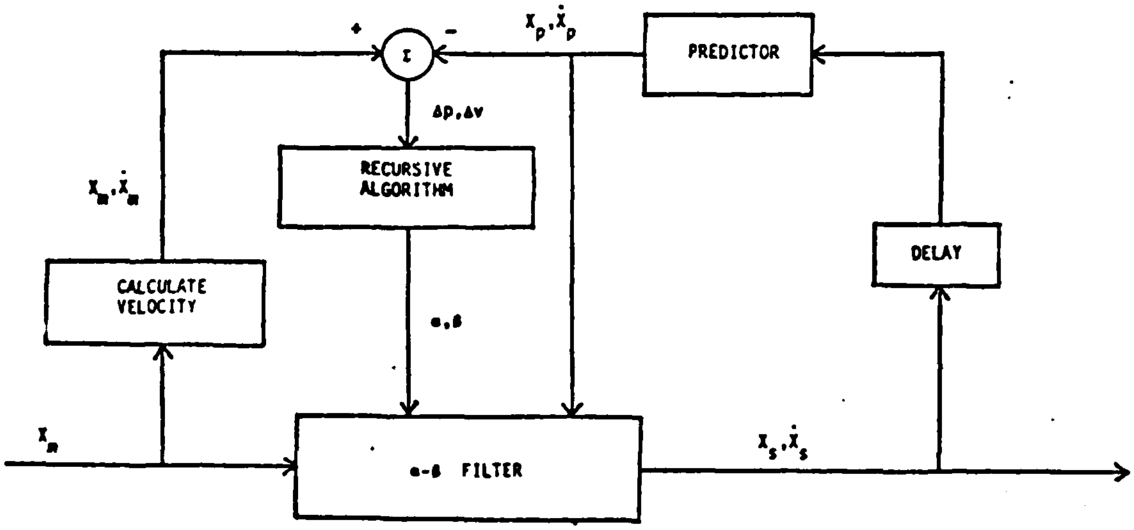


Figure 4-2. A Recursively Adaptive Filter

IV. CONCLUSIONS

Both methods of achieving better radar tracking of maneuvering targets which have been presented are being evaluated via computer simulation. Preliminary results using selected test inputs show that an improvement in performance can be achieved when compared to a fixed parameter α-β filter. Current tests involve implementing these filters in a computer simulation modeling the entire control process of the MATCALS system, however results have not yet been completely analyzed to warrant inclusion in this report.

REFERENCES

[1] T. R. Benedict and G. W. Bondner, "Synthesis of an Optimal Set of Radar Track-While-Scan Smoothing Equations," IRE Transactions on Automatic Control, Vol. AC-7, No. 4, pp. 27-32, July 1962.

[2] H. D. Helms, "Maximally Reliable Exponential Prediction Equations for Data-Rate-Limited Tracking Servomechanisms," The Bell System Technical Journal, Vol. XLIV, pp. 2343-2345, December 1965.

[3] R. E. Wilcox, "The α-β-γ Tracking Filter in the Z-Domain," IEEE National Aerospace and Electronic Conference 1979. pp. 1042-1046.

[4] C. C. Schooler, "Optimal α-β Filters for Systems with Modeling Inaccuries," IEEE Transactions on Aerospace and Electronic Systems, Vol. AES-11, No. 6, pp. 1300-1306, November 1975.

[5] "Mode I Improvement Study," ITT Gilfillan Technical Report, Contract N00039-75-C-0021, October 1978.

END

DATE
FILMED
12-81

DTIC